RES MILITARIS Social Science Journal

Analyzing Human Crowd Behavior through Video Segmentation and Classification Using Expectation-Maximization and Deep Learning Frameworks

Priti Singh, Hari Om Sharan, C.S. Raghuvanshi

Faculty of Engineering and Technology, Rama University, Mandhana, Kanpur, Uttar Pradesh, India

preetirama05@gmail.com

Abstract:

In recent years, there has been a notable surge in interest surrounding automated crowd behavior analysis. With the aim of ensuring the peaceful conduct of events and minimizing casualties in locations of public and religious significance, crowd behavior analysis has emerged as an indispensable tool worldwide. To achieve precise outcomes, it's imperative to effectively tackle the nonlinearity inherent in real-world images and videos. Deep learning-based techniques have notably advanced crowd behavior analysis, mirroring progress seen across various realms of computer vision. This study introduces a novel approach to analyzing human crowd behavior, leveraging segmentation and classification through deep learning architectures. Human crowd videos are processed to eliminate noise and extract the scene featuring the crowd. Segmentation is performed using an expectation-maximization-based ZFNet architecture, followed by classification utilizing transfer exponential Conjugate Gradient Neural Networks. Experimental evaluations are conducted across diverse human crowd datasets, assessing metrics including mean average precision, mean square error, training accuracy, validation accuracy, and specificity. The proposed technique achieves a mean average precision of 59%, mean square error of 61%, training accuracy and validation accuracy both at 95%, and specificity at 88%.

Keywords: Human crowd, behaviour analysis, ZFNet architecture, Conjugate gradient, expectation-maximization

1. Introduction:

The behaviours or actions of a group of people who have assembled for a brief time while paying attention to a specific item or event. A common component of many human endeavours is crowdedness. Every day, many pedestrians are handled in transport hubs, tall buildings, stadiums, and other public places. Effective crowd control is crucial for

maintaining safety in these situations and determining one's quality of life. Fires, crowd violence, or the ecstasy of a few crowd members are only a few examples of crowd tragedies, in which people are seriously injured or killed as a result of being crushed or trampled. Such incidents can and have happened during rock concerts, religious services, and athletic events [1]. During the admission, occupation, and evacuation of something like a public event facility, serious injury and disease can occur.Because there are so many cameras available now that make it easy to record and save video, video surveillance of individuals is an often used technology. The majority of these tools rely on a user to review the material that has been stored and interpret its content. Given this restriction, it is vital to offer video surveillance systems that enable automatic behaviour recognition [2]. Computer vision techniques can be used to implement these kinds of systems because they make it possible to recognise unsupervised patterns of human activity, such as gestures, movements, and other activities. There are numerous studies being done right now on human behaviour analysis, like [3], which have helped to identify different forms of human behaviour in video clips. Taking into account their range in time from seconds to hours, these behaviours have been ranked from the most basic to the most sophisticated. When taking into account these CCTV cameras and other installation systems, automated crowd research actually plays a significant part in crowd analysis and visual surveillance recordings [4]. Designing public areas, visual surveillance systems, and intelligently managed physical environments are so important. These kinds of systems will have many useful uses, such as crowd flow monitoring, accident management, and coordinating evacuation plans necessary in the unfortunate case of a sudden and uncontrolled fire or in the presence of riots in urban areas in particular [5]. Researchers have looked into the situation of acquiring motion data at a higher level in the research paperwork. This indicates that the motion information does not account for specific moving or stationary objects. As a result, these techniques frequently require a variety of features, such as multi-resolution histograms, spatio-temporal cuboids, appearance or motion descriptors, and spatio-temporal cubes [6].

Contribution of this research is as follows:

- To propose novel method in human crowd based behaviour analysis using segmentation and classification by DL architectures.
- Video scene has been segmented utilizing expectation-maximization based ZFNet architecture.

- The segmented video has been classified using transfer exponential conjugate gradient neural networks.

2. **Related works:**

Crowd safety in public places has always been a serious but difficult issue, especially in high-density gathering areas. The higher the crowd level, the easier it is to lose control [7], which can result in severe casualties. In order to aid in mitigation and decision-making, it is important to search out an intelligent form of crowd analysis in public areas. Crowd counting and density estimation are valuable components of crowd analysis [8], since they can help measure the importance of activities and provide appropriate staff with information to aid decision-making. As a result, crowd counting and density estimation have become hot topics in the security sector, with applications ranging from video surveillance to traffic control to public safety and urban planning [9]. Numerous crowd analysis articles were examined in work [10]. The two main subfields of crowd analysis are statistics and behaviour. Anomaly detection is frequently discussed in crowd behaviour analysis. Any subtopic of crowd behaviour analysis can experience anomalies. Finding unknown or understudied crowd analysis sub-areas that could profit from DL is the goal of this project. The author of [11] studied the crowd-related literature, including techniques for behaviour analysis and crowd surveillance. The author also provided descriptions of the methodology and datasets used. Different techniques and current deep learning concepts have been assessed. The various contemporary methods for crowd monitoring and analysis are explained in this text. An image categorization, crowd control, and warning system for the Hajj was proposed in study [12]. CNN, a DL technology, is used to classify images. Recently, the scientific and industrial sectors have become interested in several applications of CNN for speech recognition and picture classification. Density Independent and Scale Aware Model (DISAM), which author [13] suggests, performs well for high density crowds where the human head is only part visible in photos. To determine probabilities of a head in an image, CNN is first employed as a head detector as well as then utilised to compute a response matrix using scale-aware head suggestions. According to [14], the detection method known as "you only look once" (YOLO) is frequently employed to find objects in images with high levels of perspective values, or maximum threshold values. CNN and learn to scale were advised by work [15] to produce multipolar normalised density maps for crowd counting.

By using a density estimation procedure, it obtains a patch-level density map, which it then groups into different densities. Using an online learning technique for centre with multi polar loss, each patch density map is normalised. In [16], CNN as well as short term memory are utilized to calculate crowd density in surveillance videos. For estimating crowd density [19], two traditional DCN, Googlenet [17] and VGGnet [18], were utilised. Similar to this, [20] first estimates the size of the crowd in general, and then counts the precise number of persons present. The accuracy of 90% is still maintained by the efficiency. To find and keep an eye on a person in a crowded area, localisation information might be employed [21]. A regression guided detection network (RDNet) for RGB-Datasets has been developed to locate heads in images by simultaneously estimating head counts as well as localising heads using bounding boxes. Similar to [22], an accurate localization of the heads in a dense image was achieved using a density map. Using the LSC-CNN, localization was discovered in [23] with the aid of a statistic called Mean Localization Error (MLE). [24] employed image processing to determine crowd behaviour using optical flow as well as motion history image techniques. Similar to [25], anomalous behaviour identification was accomplished using an optical flow approach and Support Vector Machine (SVM). For the purpose of detecting crowd activity, Cascade Deep AutoEncoder (CDA) and combining of multi-frame optical flow information is proposed in [26]. Anomalous crowd identification was done using isometric mapping (ISOMAP), spatio-temporal, and spatio-temporal texture models.

3. System model:

In this section proposed model for video segmentation and classification based human crowd analysis with their behaviour utilizing DLtechniques. Input has been collected as surveillance video and processed for noise removal, obtain the video scene of crowd. the video scene has been segmented using expectation-maximization based ZFNet architecture. the segmented video has been classified using transfer exponential Conjugate gradient NN. Proposed architecture is shown in figure-1.

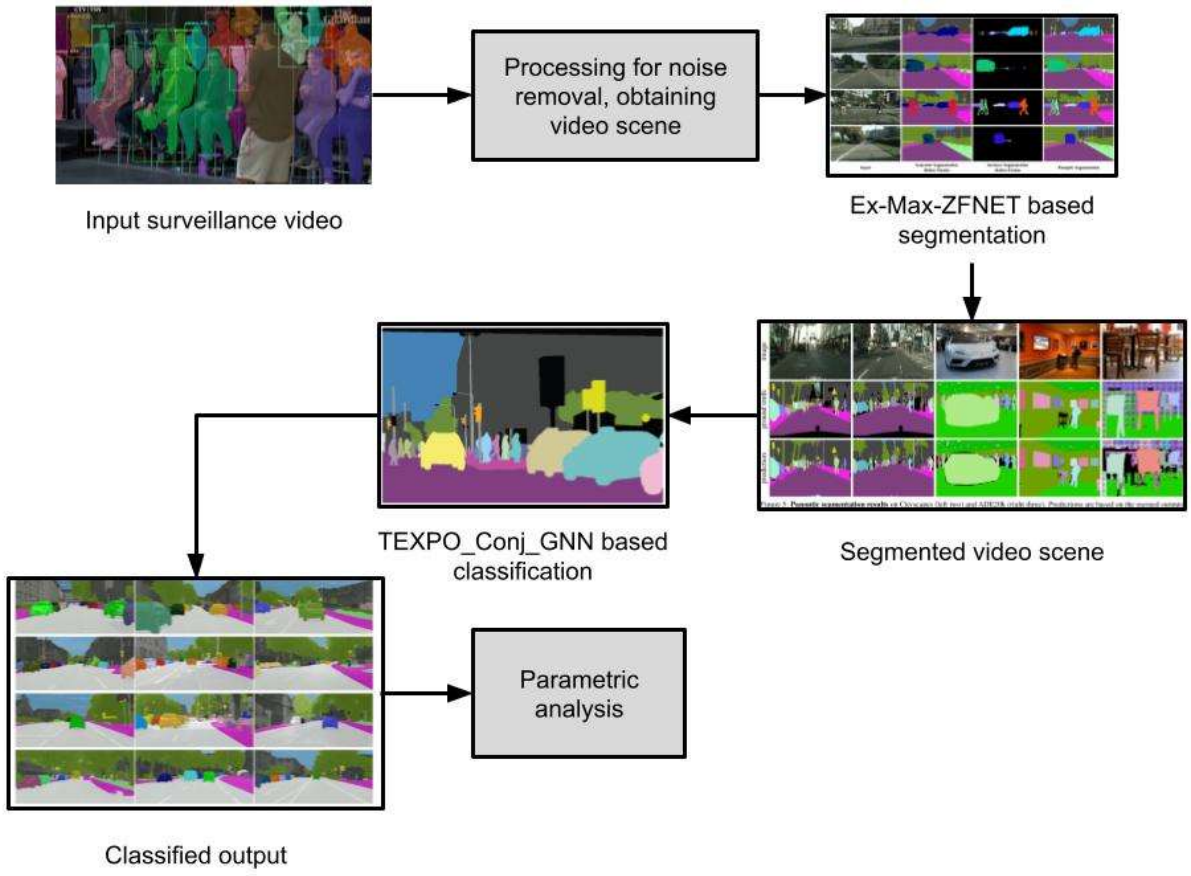

Figure 1: Proposed architecture

Data processing comprises three steps: First, the "individual activity recognition chain" is used to deduce the features of specific user actions. This processing chain converts sensor signals to time series of behavioural primitives or to quantitative user behaviour characteristics using parts of signal processing and ML. The conduct of pairs of people is then examined, producing a measure of disparity in light of presumptive crowd behaviour. Every frame scene is divided into a number of separate, nonoverlapping cubes. Global and local descriptors are used in the second part of this structure. The local descriptor, a kind of local patch descriptor, determines how similar patches are by using the structural similarity index method (SSIM) approach. As a result, two types of local descriptors based on the inner temporal approach (TIA) and space-time neighbourhood approach are carried out. Regarding the first local description, each patch's space-time neighbourhood sections consist of one for the spatial neighbourhood, which includes the patch itself in the center, and one for the temporal neighbourhood, which comes after the patch. The initial local descriptor [d0,..., d9] gives rise to the SSIM values. In terms of the TIA, the SSIM value is calculated as [D0..., Dt-

1] for each frame in the patch. Finally, the combined SSIM values from the two approaches are used to create the local descriptor [d0,..., d9, D0,...,Dt-1].

Expectation-maximization based ZFNet architecture in video segmentation:

We will now go through how the Expectation-Maximization technique was used to fit the WMM. As is customary, I is considered incomplete and is supplemented with a gdimensional z b, where z =1 is true if r i I comes from the kth component and 0 otherwise. Component memberships are defined as realisations of random vectors $z_1, z_2, \dots, z_n)$ dispersed unconditionally according to the Mulr multinomial distribution $(1, \pi_1, \dots, \pi_k)$. The EM algorithm determines maximum likelihood estimates by iteratively maximising the conditional expectation $Q\left(\Psi: \hat{\Psi}^{n)}\right.$ of the complete-data log likelihood in (1,2) with respect to the observed data v given an estimate Wb $\overset{\Psi^{n+4}}{\square}$ for the parameters.

$$= \sum_{i=1}^{s} E_{\hat{\gamma}_i}\left[\log\left\{f(z_i\Psi)f\left(\Gamma_i \mid Z_i\Psi^i\right)\right\}\Gamma_i\right]$$
$$= \sum_{i=1}^{s}\sum_{k=1}^{g} E_{\hat{p}_n}|Z_k|\Gamma_i]\log\left\{\dot{\pi}_k^{(i)}\rho W\left(\Gamma_i, \sum\nolimits^{\wedge}{}_k^{(i)}, \hat{n}_k^{(j)}\right)\right\} \qquad (1)$$

$$= \frac{\dot{\pi}_i^n f_w\left(r_i, \hat{2}_k^{(i)}, \hat{n}_i^{(n)}\right)}{\sum_{i-1}^{*} \hat{n}_i^n f_w\left(r_i, \hat{\Sigma}_i^n, \hat{n}_i^{(t)}\right)} \qquad (2)$$

As it represents an estimate of the posterior probability that Γ_i belongs to kth component of mixture under given parameter set$\hat{\psi}$, this number can be symbolised by the symbol z_i^n. The algorithm's maximum stage involves increasing $Q(\psi; \hat{\varphi}m)$by equation (3) to obtain a fresh estimate of the parameters, $\hat{\Psi}(t+1)$.

$$\hat{\Psi}^{(i+1)} = \text{argmax } Q\left(\Psi^*\hat{\Psi}^{(n}\right) \qquad (3)$$

By maximising$Q\left(\Psi_{;}\hat{\Psi}^*\right)$ with the restriction $\sum_{i=1}^{\pi}\pi_k^{\pi+1)} = 1$, the new estimates $\dot{\pi}_k^{n+1)}$ forπ_k are produced. The result is a straightforward update rule via eq (4)

$$\pi_k^{(N+1)} = \frac{1}{N}\sum_{i=1}^{N} z_k^n \qquad (4)$$

By utilising a few matrix derivation techniques. By using eq. (5), we can get the update equations for various parameters.

$$\begin{aligned} &= \sum_{i=1}^{N} \frac{\partial}{\partial \Sigma_i} (z_u^* \log \{\tilde{t}_k^n f_w(r_i, z_i, m_k)\}) \\ &= \sum_{i=1}^{N} Z_i^n \frac{\partial}{\partial \Sigma_k} \log\ f_w(r_i{:}\,\Sigma_i, n_i) \\ &= \sum_{k=1}^{N} z_k^N \left(\frac{1}{2} E_k^{-1} r_i \Sigma_k^{-1} - \frac{n_k}{2} E_i^{-1}\right) \end{aligned} \quad (5)$$

After premultiplying the previous equation by 2, we obtain the following for all k by equation (6):

$$\frac{\partial}{\partial \Sigma_k} Q\left(\Psi_{;} \overline{\mathbf{T}}^n\right) = 0 \approx \Sigma_k^{p+1)} = \frac{\sum_{i=1}^{n} \bar{z}_a^n \Gamma_1}{\sum_{*-1}^{n} \hat{z}_k^n n_k} \quad (6)$$

But it's clear that this estimator is reliant on the "values. By maximising $Q\left(\nabla_{:} \hat{\psi}^{'i}\right)$ with respect to $n_{L'}$, parameter $n_{N'}$is really first estimated in practise. This is comparable to solving the following equation(7) separately for each component

$$\sum_{i=1}^{N} z_n^n \log \left|\frac{r_i z_k}{2}\right| = \sum_{i=1}^{N} z_i^n \sum_{j=1}^{n} \psi\left(\frac{1}{2}(n_k - j + 1)\right) \quad (7)$$

where the digamma function, ψi^i is represented by the letter Σi_{is} in Eq (6). Then, formula (7) is solved mumerically in a small number of iterations, and the solution $\overline{n_e^{0+1}}$ is reintroduced in (7) to have a suitable value for 2_i^{n+11}.

An unstable segmentation network that is biassed towards the class with a wide region can result from segmentation network training with class imbalanced data. In segmentation networks, the choice of the loss functions is critical, particularly when tackling highly unbalanced issues.We offer a number of loss function types that are frequently utilised singly or in combination in networks that segment medical images. Resampling the data space is one way to solve the issue at the data-level. The most popular method for segmenting images is cross-entropy loss function.Equation(7) is utilised to calculate it. Cross-entropy loss averages all the pixels after evaluating each pixel vector's individual class predictions, which can result in some mistake if the image has an unbalanced class representation.Figure 2 depicts the ZFNet architecture.

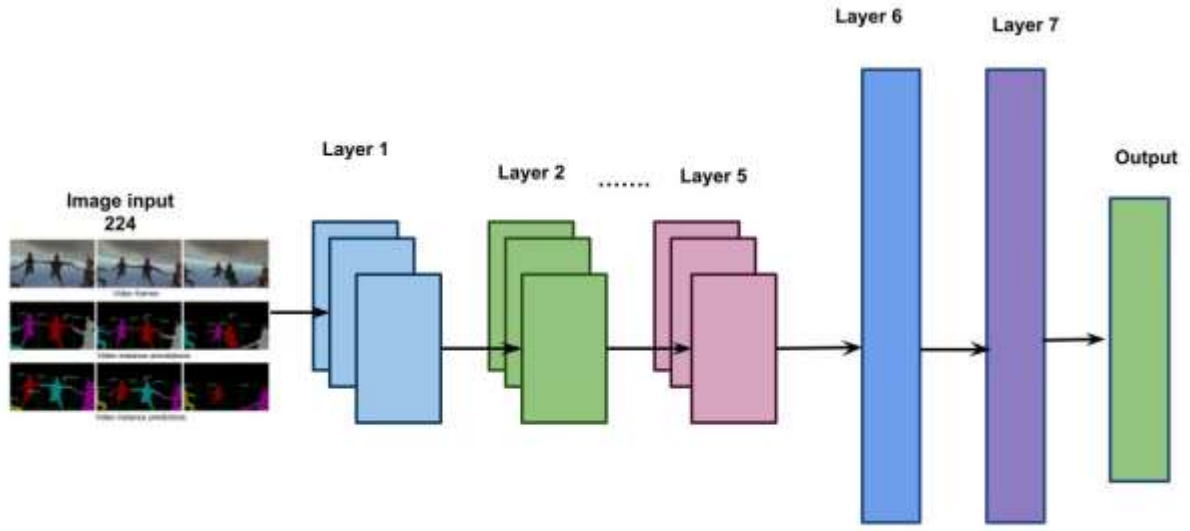

Figure 2: architecture of ZFNet

A lower number suggests tighter connection between the same object sections in multiple photos and better consistency in the change caused by the masking procedure. Utilizing features from layers l = 5 and l = 7, we compare scoresΔ for the left eye, right eye, and nose to random areas of object. The layer 5 features' lower scores for these regions compared to random object regions demonstrate that the model does build some degree of correlation.

Transfer exponential Conjugate gradient neural networks based classification:

An image with three layers—height (h), width (w), and depth (d)—serves as input data. While d is feature or channel dimension, h and w explained spatial dimension.Initial layer of image has a $h \times w$ dimension and a d colour channel (d = 1 for grayscale intensity only or d = 3 for RGB intensity). Let's say we have input data vector xij at position I of a specific layer. Following equation (eq.) (8) can also be used to calculate the vector output yij:

$$y_{ij} = f_{ks}\left(\{x_{ij} + \prec isj + sj\}, 0 \leq \delta_{i,}\delta_j \leq k\right) \qquad (8)$$

By utilising this layer, the network's parameter count can be greatly decreased. Using the equation, define $\{x_i\}_{i=1}^{n}$ as a collection of independent random variables on X with uniform distribution (9).

$$I_n(g) = \frac{1}{n}\sum_{i=1}^{n} g(\boldsymbol{x}_i) \qquad (9)$$

Then a simple evaluation gives us $\mathbb{E}(I(g)-I_n(g))^2=\frac{\text{Var}\,(g)}{n}$, Var $(g)=\int_X g^2(\boldsymbol{x})d\boldsymbol{x}-\left(\int_X g(\boldsymbol{x})d\boldsymbol{x}\right)^2$

Since the NN consits of input layer, an output layer, and a hidden layer, determining hidden layer's output is necessary before calculating the overall network output. Eq. (9), where nfindicates activation function, $\vec{\imath}$ represents the hidden neuron, idenotes the input neurons, and $mm_{(ai)}$ signifies bias weight, is used to determine the hidden layer output, or ein. Equation(10) gives the NN model

$$\begin{aligned} e^{(n)} &= nf\left(w_{(ni)}^{(m)}+\sum_{j=1}^{n} w_{(j)}^{(N)}F_D\right) \\ \hat{\sigma}_{\hat{o}} &= nf\left(w_{(\omega\dot{o})}+\sum_{i=1}^{s} w_{(\hat{\imath})}^{(\omega)}e^{(m)}\right) \end{aligned} \qquad (10)$$

The weight matrices provided in (9) and (10) are used to generate weight space and accompanying biases for ERN optimization, as shown in eq. (11):

$$\begin{aligned} &W_n=U_n=\sum_{m=1}^{N} a\cdot\left(\text{rand}\ -\frac{1}{2}\right). \\ &B_n=\sum_{n=1}^{N} a\cdot\left(\text{rand}\ -\frac{1}{2}\right). \\ &\left|\mathscr{R}(f)-\hat{\mathscr{R}}_n(f)\right|\leq \sup_{f\in\mathscr{H}_m}\left|\mathscr{R}(f)-\hat{\mathscr{R}}_n(f)\right|=\sup_{f\in\mathscr{H}_m}|I(g)-I_n(g)| \end{aligned} \qquad (11)$$

where $W_n=N$ weight in the weight matrix. The random number in (1) is called the rand, and it falls between [0,1], where an is any constant parameter for suggested approach, and it is less than 1, and Bn is a bias value. Consequently, weight list matrix is provided in equation (12):

$$W^c=[W_n^1,W_n^2,W_n^3,\dots,W_n^{N-1}]. \qquad (12)$$

Sum of square mistakes may now be readily anticipated for each weight matrix from the neural network procedure. One input layer, one hidden or "state" layer, and one "output" layer make up three-layer network employed for the ERN structure.

$$\begin{aligned} &\text{net}_j\,(t)=\sum_i^n x_i(t)w_{m(j)}+B_{m(j)} \\ &\inf_{m\in K_m}\|f^*-f_m\|_{L^2(P)}^2\lesssim\frac{\Delta(f^*)^2}{m} \end{aligned} \qquad (13)$$

where Bn(j) is a bias and n is number of inputs. The input vector is similarly propagated through a weight layer in a simple recurrent network, but it is also paired with the activation of the previous state by a second recurrent weight layer, U by eq (14),

$$\begin{gathered} y_j(t) = f\big(\mathrm{net}_j\,(t)\big). \\ \mathrm{net}_j\,(t) = \sum_i^{\sum_i} x_i(t)W_{s(m)} + \sum_i^n n(t-1)U_{n(j)} + B_{m(j)}, \\ y_j(t) = f\big(\mathrm{net}_j\,(t)\big), \\ \Delta(f) := \inf_j \int_{\mathbb{R}^d} \parallel \omega \parallel_1 |\hat{f}(\omega)| d\omega < \infty, \end{gathered} \tag{14}$$

where f is an extension of f to $L2(Rd)$ Fourier transform. In (14) the convergence rate is dimension-independent. However, because it uses the Fourier transform, constant $\Delta(f*)$ could be dimension-dependent.

In both scenarios, network's output is governed by state and a collection of output weights W by eq (15)

$$\begin{gathered} \mathrm{net}_k\,(t) = \sum_j^M y_j(t)W_{makj} + B_{m|k}, \\ Y_k(t) = g(\,\mathrm{net}_{\,k}(t)), \end{gathered} \tag{15}$$

g is an output function. Thus, error is determined using equation (16):

$$E = (T_k - Y_k). \tag{16}$$

Equation (17) gives the network's performance index:

$$\begin{gathered} V(x) = \frac{1}{2}\sum_{k=1}^K (T_k - X_k)^T (T_k - Y_k) \\ V_F(x) = \frac{1}{2}\sum_{k=1}^K E^T \cdot E. \end{gathered} \tag{17}$$

$$V_\mu(x) = \frac{\sum_{j=1}^N V_F(x)}{P_i}. \tag{18}$$

A random feature method is given byeq. (19)

$$f_m(\boldsymbol{x};\boldsymbol{a}) = \frac{1}{m}\sum_{j=1}^m a_j \phi\big(\boldsymbol{x};\boldsymbol{w}_j^0\big) \tag{19}$$

where the i.i.d random variables w0j and {w0j}m$j = 1$ are selected from the prefixed distribution 0 The coefficients are $a = (a1, \ldots, am)T \in R$m and the collection is

{$\varphi(;;w0j)$}are the random characteristics. The replicating kernel Hilbert space (RKHS), which is caused by the kernel by eq, is the natural function space for this paradigm (20)

$$k(\boldsymbol{x}, \boldsymbol{x}') = \mathbb{E}_{\boldsymbol{w}\sim\pi_0}[\phi(\boldsymbol{x};\boldsymbol{w})\phi(\boldsymbol{x}';\boldsymbol{w})] \qquad (20)$$

Denote by Hk this RKHS. Then for any $f \in$ Hk, there exists $a(\cdot) \in$ L2(π0) such thateq. (21)

$$\begin{aligned} f(\boldsymbol{x}) &= \int a(\boldsymbol{w})\phi(\boldsymbol{x};\boldsymbol{w})d\pi_0(\boldsymbol{w}), \\ \| f \|^2_{\mathcal{H}_k} &= \inf_{a\in S_f} \int a^2(\boldsymbol{w})d\pi_0(\boldsymbol{w}), \end{aligned} \qquad (21)$$

Variations in batch-wise training come from the gradient variance. The use of a random sample has the advantage of requiring much fewer computations per iteration while the disadvantage is the noisy gradient.Please take note that iterations are used to calculate the convergence rate in this section. We must first define the Lyapunoy process using equation (22) in order to study the training dynamics each iteration.

$$h_t = \|\mathbf{w}^t - \mathbf{w}^*\|_2^2 \qquad (22)$$

The formula calculates the separation between the existing solution, wt, and the ideal solution, w * ht, where ht is a random variable. As a result, using equation (23), one can determine the SGD's convergence rate:

$$\begin{aligned} h_{t+1} - h_t &= \|\mathbf{w}^{t+1} - \mathbf{w}^*\|_2^2 - \|\mathbf{w}^t - \mathbf{w}^*\|_2^2 \\ &= (\mathbf{w}^{t+1} + \mathbf{w}^{\mathbf{t}} - 2\mathbf{w}^*)(\mathbf{w}^{t+1} - \mathbf{w}^{\mathbf{t}}) \\ &= \left(2\mathbf{w}^{\mathbf{t}} - 2\mathbf{w}^* - \eta_t \nabla\psi_{\mathbf{w}}(\mathbf{d_t})\right)\left(-\eta_t \nabla\psi_{\mathbf{w}}(\mathbf{d_t})\right) \\ &= -2\eta_t(\mathbf{w}^t - \mathbf{w}^*)\nabla\psi_{\mathbf{w}}(\mathbf{d_t}) + \eta_t^2\left(\nabla\psi_{\mathbf{w}}(\mathbf{d_t})\right)^2 \end{aligned} \qquad (23)$$

It is a random sample of d in the sample space Ω, and the random variable ht+1 - ht depends on the sample drawn (dt) and the rate of learning (nt). It indicates the extent to which reducing YAR{∇yw(dt)} improves the convergence rate.We gauge SGD's effectiveness using $R(k) = \mathbb{E}\left[\|z(k) - z^*\|^2\right]$, which represents the anticipated squared distance between solution at time k and ideal solution. We will focus on two error terms, which is different from analysis for SGD.The expected squared distance between z(k) and z* is defined by the first term, termed the expected optimization error. Equation(24) provides average squared distance between each iterate's zi(k) and ideal z*

$$\frac{1}{n}\sum_{i=1}^{n}\mathbb{E}\left[\|z_i(k)-z^*\|^2\right]=\mathbb{E}\left[\|\bar{z}(k)-z^*\|^2\right]+\frac{1}{n}\sum_{i=1}^{n}\mathbb{E}\left[\|z_i(k)-\bar{z}(k)\|^2\right]. \quad (24)$$

Therefore, examining the two terms will give us information about how DSGD is performing. Denote with by eq(25) to make notation simpler

$$U(k)=\mathbb{E}\left[\|\bar{z}(k)-z^*\|^2\right], V(k)=\sum_{i=1}^{n}\mathbb{E}\left[\|z_i(k)-\bar{z}(k)\|^2\right], \forall k \quad (25)$$

We decided to use eq(26) after being motivated by the SGD analysis

$$U(k+1)\leq\left(1-\frac{1}{k}\right)^2 U(k)+\frac{2L}{\sqrt{n}\mu}\frac{\sqrt{U(k)V(k)}}{k}+\frac{L^2}{n\mu^2}\frac{V(k)}{k^2}+\frac{\sigma^2}{n\mu^2}\frac{1}{k^2} \quad (26)$$

reflecting the additional disturbances brought on by the variations in solutions, the observed consensus error V(k). Relation (17) also shows that projected convergence rate of U(k) for SGD cannot be better than R(k). However, if V(k) decays quickly enough relative to U(k), it is likely that two extra terms will be insignificant over time, and we would assume that U(k) will converge at a pace similar to R(k) for SGD.

4. Performance analysis:

A PC running Windows 7 64-bit with an Intel Xeon E5-1650 served as the training platform. The software tools consisted of Microsoft Visual Studio 12.0, Python 2.7, CUDNN 7.5, and CUDA 8.1.

Dataset description: The first dataset used for population counts was from UCSD. The information was gathered using a camera that was mounted on a walkway for pedestrians. Dataset consists of 2000 frames of video sequences with a resolution of 238 × 158 pixels, combined with ground truth annotation of every fifth frame's 49,885 pedestrians. Mall dataset was gathered using security cameras placed up at a shopping mall. 2000 frames total, each measuring 320 x 240. The difficult UCF CC 50 dataset contains a wide range of densities and different sceneries. This information was collected from a variety of locations, including stadiums, marathons, political rallies, and concerts. There are 50 annotated photographs in all, with an average of 1279 people per image. Individuals in this dataset range in resolution from 94 to 4543, indicating a wide range in the image. The limitation on the number of photos available for training and evaluation is a downside of this type of dataset. This dataset's 220 maximum crowd count is too low to accurately assess counting of highly dense crowds. The Shanghai Tech dataset, which consists of 1198 photos with 330,165 labelled

heads, has been released for large-scale crowd counting. This collection is one of the largest in terms of annotated heads. There are two categories in the dataset: Part A and Part B. Part A contains 482 randomly selected photographs from internet. While Part B contains 716 photos that were gathered from a Shanghai city street. UCF-QNRF, which contains 1535 pictures, is the most recent dataset. The range of individuals in this dataset, from 49 to 12,865, results in a significant fluctuation in population density. Furthermore, it includes crowd movies with a range of crowd densities and perspective scales and has a huge image resolution from 400 × 300 to 9000 × 6000. The CUHK dataset was gathered in a variety of places, including streets, malls, airports, and parks. 474 video clips from 215 scenes make up the dataset shown in Table 1.

Table 1: Description of datasets

Datasets	**Description**	**No.of images**	**Resolutions**	**Min**	**Ave**	**Max**	**Overall count**	**Accessibility**
UCSD	People counting	2000	238x158	11	25	46	49,885	Yes
MALL	People counting	2000	320x240	13	-	53	62,325	Yes
UCF_CC_50	Density estimation	50	Variable	94	1279	4543	63,325	Yes
World Expo 10	Cross scene crowd counting	3980	576x720	1	50	253	199,923	Yes
Shanghai Tech A,B	Crowd counting	482	Variable	33	501	3139	241,677	Yes
UCF-QNRF	Crowd counting	716	400x300 to	9	123	578	88,488	yes

	and localization		9000x6000					
CUHK	Crowd behaviour	1535	Variable	49	815	12,865	-	Yes

Table 2: Analysis for various video dataset based on human crowd behaviour

Datasets	**Techniques**	**MAP**	**MSE**	**Training accuracy**	**Validation accuracy**	**Specificity**
UCSD	**CNN**	41	38	68	72	65
	SVM	43	42	72	74	68
	HCB_VSC_DLA	44	43	75	77	71
MALL	**CNN**	42	39	72	75	69
	SVM	46	45	73	77	73
	HCB_VSC_DLA	49	47	75	79	75
UCF_CC_50	**CNN**	44	41	74	79	71
	SVM	48	43	78	85	76
	HCB_VSC_DLA	51	48	81	88	77
World Expo 10	**CNN**	45	44	78	81	73
	SVM	49	48	79	83	77
	HCB_VSC_DLA	53	49	83	86	79
Shanghai Tech A,B	**CNN**	47	48	81	83	75
	SVM	49	52	83	88	81
	HCB_VSC_DLA	53	53	85	89	83
UCF-QNRF	**CNN**	49	51	82	85	81
	SVM	51	53	85	89	83
	HCB_VSC_DLA	53	55	88	92	85
CUHK	**CNN**	52	55	84	91	82
	SVM	55	58	89	93	86
	HCB_VSC_DLA	59	61	95	95	88

Table-2 analysis for various video dataset based on human crowd behaviour. the dataset compared are UCSD, MALL, UCF_CC_50, World Expo 10, Shanghai Tech A,B, UCF-

QNRF, CUHK. the parameters analysed areMAP, MSE, training accuracy, validation accuracy, specificity.

5. Conclusion:

In this study, we analyze human crowd behavior through video segmentation and classification. Surveillance videos are collected and processed to extract video frame-based scenes. These scenes are then segmented using an expectation-maximization-based ZFNet architecture and classified using transfer exponential Conjugate Gradient Neural Networks. Experimental results on a real human activity database illustrate that deep learning (DL) surpasses both data mining and state-of-the-art methods in terms of both runtime and accuracy performance. We evaluate the proposed methods using multiple authentic human behavioral databases, achieving a Mean Average Precision (MAP) of 59%, Mean Squared Error (MSE) of 61%, training accuracy of 95%, validation accuracy of 95%, and specificity of 88%.

Reference:

1. Tyagi, B., Nigam, S., & Singh, R. (2022). A review of deep learning techniques for crowd behavior analysis. *Archives of Computational Methods in Engineering*, *29*(7), 5427-5455.
2. Chaudhary, D., Kumar, S., & Dhaka, V. S. (2022). Video based human crowd analysis using machine learning: a survey. *Computer Methods in Biomechanics and Biomedical Engineering: Imaging & Visualization*, *10*(2), 113-131.
3. Bruno, A., Ferjani, M., Sabeur, Z., Arbab-Zavar, B., Cetinkaya, D., Johnstone, L., ... &Benaouda, D. (2022, August). High-level feature extraction for crowd behaviour analysis: a computer vision approach. In *Image Analysis and Processing. ICIAP 2022 Workshops: ICIAP International Workshops, Lecce, Italy, May 23–27, 2022, Revised Selected Papers, Part II* (pp. 59-70). Cham: Springer International Publishing.
4. Kong, Y. X., Wu, R. J., Zhang, Y. C., & Shi, G. Y. (2023). Utilizing statistical physics and machine learning to discover collective behavior on temporal social networks. *Information Processing & Management*, *60*(2), 103190.

5. Farooq, M. U., Mohamad Saad, M. N., Saleh, Y., & Daud Khan, S. (2022, November). Deep Learning Approach for Divergence Behavior Detection at High Density Crowd. In *International Conference on Artificial Intelligence for Smart Community: AISC 2020, 17–18 December, UniversitiTeknologi Petronas, Malaysia* (pp. 875-888). Singapore: Springer Nature Singapore.
6. Sharma, V., Mir, R. N., & Singh, C. (2023). Scale-aware CNN for crowd density estimation and crowd behavior analysis. *Computers and Electrical Engineering*, *106*, 108569.
7. Bahamid, A., &Mohd Ibrahim, A. (2022). A review on crowd analysis of evacuation and abnormality detection based on machine learning systems. *Neural Computing and Applications*, *34*(24), 21641-21655.
8. Bhuiyan, M. R., Abdullah, J., Hashim, N., & Al Farid, F. (2022). Video analytics using deep learning for crowd analysis: a review. *Multimedia Tools and Applications*, 1-28.
9. Matkovic, F., Ivasic-Kos, M., &Ribaric, S. (2022). A new approach to dominant motion pattern recognition at the macroscopic crowd level. *Engineering Applications of Artificial Intelligence*, *116*, 105387.
10. Hou, H., & Wang, L. (2022). Measuring Dynamics in Evacuation Behaviour with Deep Learning. *Entropy*, *24*(2), 198.
11. Pattan, P., &Arjunagi, S. (2022). A human behavior analysis model to track object behavior in surveillance videos. *Measurement: Sensors*, *24*, 100454.
12. Abpeikar, S., Kasmarik, K., Garratt, M., Hunjet, R., Khan, M. M., &Qiu, H. (2022). Automatic collective motion tuning using actor-critic deep reinforcement learning. *Swarm and Evolutionary Computation*, *72*, 101085.
13. Zhang, D., Li, W., Gong, J., Huang, L., Zhang, G., Shen, S., ... & Ma, H. (2022). HDRLM3D: A Deep Reinforcement Learning-Based Model with Human-like Perceptron and Policy for Crowd Evacuation in 3D Environments. *ISPRS International Journal of Geo-Information*, *11*(4), 255.
14. Lu, Y., Ruan, X., & Huang, J. (2022). Deep Reinforcement Learning Based on Social Spatial–Temporal Graph Convolution Network for Crowd Navigation. *Machines*, *10*(8), 703.
15. Liu, T., Zheng, Q., & Tian, L. (2022). Application of Distributed Probability Model in Sports Based on Deep Learning: Deep Belief Network (DL-DBN)

Algorithm for Human Behaviour Analysis. *Computational Intelligence and Neuroscience*, *2022*.

16. Ha, D., & Tang, Y. (2022). Collective intelligence for deep learning: A survey of recent developments. *Collective Intelligence*, *1*(1), 26339137221114874.
17. Liang, Z., Li, L., & Wang, L. (2022, December). Crowd-Oriented Behavior Simulation: Reinforcement Learning Framework Embedded with Emotion Model. In *Artificial Intelligence: Second CAAI International Conference, CICAI 2022, Beijing, China, August 27–28, 2022, Revised Selected Papers, Part III* (pp. 195-207). Cham: Springer Nature Switzerland.
18. Choi, T., Pyenson, B., Liebig, J., &Pavlic, T. P. (2022). Beyond tracking: using deep learning to discover novel interactions in biological swarms. *Artificial Life and Robotics*, *27*(2), 393-400.
19. Poon, K. H., Wong, P. K. Y., & Cheng, J. C. (2022). Long-time gap crowd prediction using time series deep learning models with two-dimensional single attribute inputs. *Advanced Engineering Informatics*, *51*, 101482.
20. Tiwari, R. G., Yadav, S. K., Misra, A., & Sharma, A. (2022). Classification of Swarm Collective Motion Using Machine Learning. In *Human-Centric Smart Computing: Proceedings of ICHCSC 2022* (pp. 173-181). Singapore: Springer Nature Singapore.
21. Chakole, P. D., Satpute, V. R., &Cheggoju, N. (2022, May). Crowd behavior anomaly detection using correlation of optical flow magnitude. In *Journal of Physics: Conference Series* (Vol. 2273, No. 1, p. 012023). IOP Publishing.
22. Guo, B., Liu, Y., Liu, S., Yu, Z., & Zhou, X. (2022). CrowdHMT: Crowd Intelligence with the Deep Fusion of Human, Machine, and IoT. *IEEE Internet of Things Journal*, *9*(24), 24822-24842.
23. Tripathi, S. K. (2022). *Design and development of some methods and models for crowd analysis using computer vision and deep learning approaches* (Doctoral dissertation, IIT (BHU), Varanasi).
24. Lalit, R., &Purwar, R. K. (2022). Crowd Abnormality Detection Using Optical Flow and GLCM-Based Texture Features. *Journal of Information Technology Research (JITR)*, *15*(1), 1-15.
25. Pai, A. K., Chandrahasan, P., Raghavendra, U., &Karunakar, A. K. (2022). Motion pattern-based crowd scene classification using histogram of angular deviations of trajectories. *The Visual Computer*, 1-11.

26. Bala, B., Kadurka, R. S., &Negasa, G. (2022). Recognizing Unusual Activity with the Deep Learning Perspective in Crowd Segment. In *A Fusion of Artificial Intelligence and Internet of Things for Emerging Cyber Systems* (pp. 171-181). Springer, Cham.