Since January 2020 Elsevier has created a COVID-19 resource centre with free information in English and Mandarin on the novel coronavirus COVID-19. The COVID-19 resource centre is hosted on Elsevier Connect, the company's public news and information website.

Elsevier hereby grants permission to make all its COVID-19-related research that is available on the COVID-19 resource centre - including this research content - immediately available in PubMed Central and other publicly funded repositories, such as the WHO COVID database with rights for unrestricted research re-use and analyses in any form or by any means with acknowledgement of the original source. These permissions are granted for free by Elsevier for as long as the COVID-19 resource centre remains active.
A novel algorithm for mask detection and recognizing actions of human

Puja Gupta\textsuperscript{a,}\textsuperscript{*}, Varsha Sharma\textsuperscript{a}, Sunita Varma\textsuperscript{b}

\textsuperscript{a} School of Information Technology, Rajiv Gandhi Proudyogiki Vishwavidyalaya, Bhopal, Madhya Pradesh 462003 India
\textsuperscript{b} Department of Information Technology, Shri Govindram Seksaria Institute of Technology and Science, Indore, Madhya Pradesh 452003 India

\section*{ARTICLE INFO}

\textbf{A B S T R A C T}

Face recognition has become a significant challenge today since an increasing number of individuals wear masks to avoid infection with the novel coronavirus or Covid-19. Due to its rapid proliferation, it has garnered growing attention. The technique proposed in this chapter seeks to produce unconstrained generic actions in the video. Conventional anomaly detection is difficult because computationally expensive characteristics cannot be employed directly, owing to the necessity for real-time processing. Even before activities are completely seen, they must be located and classified. This paper proposes an expanded Mask R-CNN (Ex-Mask R-CNN) architecture that overcomes these issues. High accuracy is achieved by using robust convolutional neural network (CNN)-based features. The technique consists of two steps. First, a video surveillance algorithm is employed to determine whether or not a human is wearing a mask. Second, Multi-CNN forecasts the frame’s suspicious conventional abnormality of people. Experiments on tough datasets indicate that our approach outperforms state-of-the-art online traditional detection of anomaly systems while maintaining the real-time efficiency of existing classifiers.

\section*{1. Introduction}

Numerous strategies have been attempted to control the spread of coronavirus in view of the increasing number of cases. Additionally, Artificial Intelligence (AI) is frequently employed in this activity. Keeping a social distance and wearing masks have become fashionable. However, in order to achieve rapid results and leverage the potential of AI, certain variables were compromised, and important capabilities of AI were not exploited. The World Health Organization (WHO) has declared mask use mandatory during the COVID-19 pandemic (McIntosh, Hirsch, & Bloom, 2020). In this Covid-19 era, people’s use of face masks in public places has risen. The older face mask was worn on a temporary basis or for personal reasons; scientists have noted an increase in the mask’s use during Covid-19 (Feng, Shen, Xia, Song, Fan, & Cowling, 2020; Pradhan, Biswasroy, Kumar Naik, Ghosh, & Rath, 2020).

This epidemic has resulted in unprecedented levels of scientific cooperation throughout the world. Machine learning and deep learning algorithms powered by artificial intelligence may play a significant role in combating this pandemic. Machine learning techniques facilitate the evaluation of the huge amount of data in the Covid-19 issue for its investigation (Loey, Manogaran, Taha, & Khalifa, 2021).

To address and detect suspicious persons through observation of their behaviour, developing technology such as artificial intelligence (AI), the Internet of Things (IoT), big data, and machine learning is required (Phule & Sawant, 2017).

In this paper, we present a method for detecting face masks based on Mask R-CNN ROI wrapping with Resnet-152 and then evaluate the proposed model using Apache MXNet. Additionally, we devised an algorithm for detecting anomalous human behaviour. The model presented in this manuscript can also be used in conjunction with video surveillance systems to detect individuals wearing masks. Our work analyses algorithms with high accuracy and the shortest running time for system training and recognition.

We examine mask detection in this research using the proposed technique, which identifies whether or not persons are wearing face masks using real images or videos collected by IoT-based cameras. Following that, using standard anomaly detection, the proposed ExMask R-CNN method is applied to detect humans. The COVID-19 pandemic serves as the impetus for this endeavour due to increased social gatherings during COVID-19.

We proposed Ex-Mask R-CNN to verify human face masks and perform conventional anomaly detection using real image or videos. The following are the paper’s primary contributions:
1. To recognize human face masks utilizing image and video datasets that assist in locating individuals who are not wearing a mask during the COVID-19 epidemic.

2. The approach for detecting human face masks is based on Mask R-CNN ROI wrapping with Resnet-152.

3. To identify standard anomaly detection techniques on image and video datasets that can aid in social distancing during COVID-19.

4. The conventional way of detecting anomalies is by the use of an optical-flow stacked difference image.

The novelty of our approach is that it combines end-to-end feature extraction with machine learning algorithms for recognizing facial masks. The paper is structured as follows: Section 2 summarizes the literature reviewed. The model described in Section 3 is illustrated in Section 4. Section 4 describes and analyses the experiments, and Section 6 concludes with a discussion of prospective future work.

2. Related works

The purpose of this paper is to enable automatic action recognition in surveillance systems in order to aid people in alerting, retrieving, and summarising data (Parwez, Rawat, & Garuba, 2017; Shahroudy, Liu, Ng, & Wang, 2016; Vu, Li, Law, & Zhang, 2018). Our work utilizes the CNN algorithm to detect actions in video surveillance systems. Previously introduced approaches detected actions by the use of silhouette or form algorithms to detect actions in video surveillance systems. Previously, approaches detected actions by the use of silhouette or form algorithms to detect actions in video surveillance systems. However, to our knowledge, no technique is capable of accurately detecting action involving people wearing masks in a real-time context. Nonetheless, there is no uniform methodology that assesses the security risk associated with people wearing masks.

The proposed approach for action detection is based on a two-stage framework. However, the crucial distinction is that their network is optimized for a single actor and produces inaccurate predictions for numerous actors. Additionally, appearance-based temporal feature integration is extremely distinct, and our suggested approach is capable of detecting the activities of numerous actors who are wearing masks. We offer a method for detecting face masks that are based on Mask R-CNN ROI wrapping with Resnet-152. Below is the mathematical formulation for our suggested method based on Resnet-152 (Yin, Li, & Wang, 2019).

**ResNet with Path-Integral Formula (Yin et al., 2019).**

\[ u(x_t) = \text{relu}(f \cdot u(x_{t-1}) + \theta(x_t) \ast u(x_{t-1})) \]  

Given \( f \cdot u + \theta \ast u = f \left( \delta + \sum_{p_i} \right) \cdot u \), Eq. (1) can be rewritten as follows.

\[ u(x_t) = \sum_{k_0} kof(\delta(x_t - x_{t-1}) + \Omega(x_t - x_{t-1})) u(x_{t-1}) \]  

In Eq. (2), \( \Omega(x) = \frac{\delta(x)}{\pi} \). Under the influence of \( \text{relu} \), \( k \) deployed on \( f \) leads to \( f \) or 0 depending on \( u(x_t) \). As such, we can define \( \text{re} = \log(kof) \).

With inverse discrete Fourier transform, we can obtain functions in the frequency domain as follows.

\[ \delta(x_t - x_{t-1}) = \frac{1}{M} \sum_{k=0}^{M-1} \delta(e^{ik\pi}) \sum_{p_1} e^{ik}
\]

The drawback of MHI is that it cannot record sequences. Motion Energy Images (MEI) and Motion History Images (MHI) introduced approaches detected actions by the use of silhouette or form algorithms to detect actions in video surveillance systems. Previously introduced approaches detected actions by the use of silhouette or form algorithms to detect actions in video surveillance systems. However, to our knowledge, no technique is capable of accurately detecting action involving people wearing masks in a real-time context. Nonetheless, there is no uniform methodology that assesses the security risk associated with people wearing masks.

The novelty of our approach is that it combines end-to-end feature extraction with machine learning algorithms for recognizing facial masks. The paper is structured as follows: Section 2 summarizes the literature reviewed. The model described in Section 3 is illustrated in Section 4. Section 4 describes and analyses the experiments, and Section 6 concludes with a discussion of prospective future work.
\[ u(x_i) = \sum_{k=1}^{N} \sum_{j=1}^{M} e^{ip_{i-1}(x_k-x_{i-1})-\varphi_{i-1}} u(x_{i-1}) \]  

(7)

This corresponds to Eq. (15), where the short time propagator \( K_{t_0:t} \) is defined. In essence, \( H \) defined the energy of a specific path. After \( N \) residual convolution steps, the outputs from the network can be formulated as,

\[ u(x_i) = \prod_{p=1}^{\infty} \sum_{k=1}^{M} e^{ip_{p-1}(x_k-x_{i-1})-\varphi_{p-1}} u(x_{i-1}) \]

\[ = \sum_{\text{path}} \sum_{p=1}^{\infty} e^{ip_{p-1}(x_k-x_{i-1})-\varphi_{p-1}} u(x_{i-1}) \]  

(8)

In Eq. (8), we define \( \prod_{t=0}^{t_{n-1}} = \sum_{\text{path}} \prod_{t=1}^{n} \), where \( \sum_{\text{path}} = \sum_{k_1} \sum_{k_2} \cdots \sum_{k_i} \). This implies that summing along every path leads to \( x_k \). More strictly speaking, \( \sum_{\text{path}} \) is the functional integral over trajectory functions. Analogously, \( \sum_{\text{path}} \) can be derived. Eq. (8) can be equivalently written as follows,

\[ u(x_n) = \sum_{\text{path}} \sum_{p=1}^{\infty} e^{ip_{p-1}(x_k-x_{i-1})-\varphi_{p-1}} u(z_i) \]  

(9)

By comparing Eq. (9) to Eq. (16), Eq. (9) can be regarded as the phase space path integral of ResNet. Regarding the mathematical equivalence between residual convolution and PDE as discussed in Eq. (18), the residual convolution results in \( e^{ip_{i-1}} \) in the frequency domain. Obviously, \( \hat{T}_p = -\frac{1}{2}p^2 + ibp - c \) corresponds to \( \hat{H}(p_{i-1}) \):

\[ H(p_{i-1}) = \frac{1}{2}p^2 - ibp_{i-1} - c \]  

(10)

The two order form of Hamiltonian \( H \) guarantees that it is integral by inverse Fourier transform over frequency \( p \), such that an integral path formula in position space can be obtained.

\[ u(x_0) = \sum_{\text{path}} \sum_{\text{path}} e^{ip_{1}(x_k-x_{i-1})-\varphi_{1}} u(x_0) \]

\[ = \sum_{\text{path}} \sum_{p=1}^{\infty} e^{ip_{i-1}(x_k-x_{i-1})-\varphi_{i-1}} u(x_{i-1}) \]  

\[ = \sum_{\text{path}} \sum_{p=1}^{\infty} e^{ip_{i-1}(x_k-x_{i-1})-\varphi_{i-1}} u(z_i) \]  

(11)

By defining \( x = x_i - x_{i-1} \), with the definition of kinetic energy \( T \) and potential energy \( V \), the Lagrangian \( L = T-V \) can be obtained.,

\[ V = c - \log(\sigma) \]  

(12)

\[ T(x) = \left( x + \frac{b}{2} \right)^2 \]  

(13)

As such, the evolution of ResNet can be written based on the form of integrals over action \( S \).

\[ u(x_i) = \sum_{\text{path}} \sum_{\text{path}} e^{iT(x)} u(x_0) \]

\[ = \sum_{\text{path}} e^{i\sum_{x_{i-1}}^{i} V-T(x)} u(x_0) \]

\[ = \sum_{\text{path}} e^{\sum_{x_{i-1}}^{i} V} u(x_0) \]  

(14)

This formulation is equivalent to the Feynman path integral formulation in Eq. (17). It is regarded as the path integral formulation of ResNet, which helps us better understand the ResNet architecture. The ResNet output is given by adding the contributions along all paths that information flow through together. The contribution of a path is proportional to \( S_{\text{path}} \), where \( S_{\text{path}} \) does the time integral of the Lagrangian give the action \( L_{\text{path}} \) along the path. Lagrangian \( L_{\text{path}} \) is defined based on the kinetic energy \( T(x) \) and potential energy \( V(x) \), i.e., \( L_{\text{path}} = T(x) - V(x) \).

\[ \langle x_{i+1}|H(x_i) = \int dp \langle x_{i+1}|p_i>|p_i|H(x_i) \]  

\[ = \int dp \langle x_{i+1}|p_i|\langle p_i|H(x_i) \]  

\[ = \int dp \frac{dp}{2\pi} e^{i\varphi_{i-1}} H(p_{i-1}) \]  

(15)

\[ K = \int D_{\alpha}(t) e^{i\int dt \varphi_{i-1}} \]  

(16)

\[ \bar{u}(p, t) = e^{\frac{i\pi}{2} T(p)} \]  

(17)

The first to eighteenth equations are for the Feynman path integral from ResNet. The equation demonstrates how ResNet mathematically works. As a result of the Feynman route integral formulation above, we may determine the end state by combining the contributions of all pathways in the configuration space. The contribution of a path is proportional to \( e^{iS} \), where \( S \) is the action given by the time integral of the Lagrangian \( L \) along the path.

2.1. Problem identification

The following work has been identified after a deep study of literature related to testing and training environment, Dataset name, Dataset for training, Dataset for testing.

a. Testing and training environment

Python Programming Language is used to design 15.6 HD WLED touch screen (1366 × 768), 10-finger multi-touch support. 10th Generation Intel Core i7-1065G7 1.3 GHz up to 3.9 GHz. 8 GB DDR4 SDRAM 2666 MHz, 512 GB SSD, No Optical Drive, Intel Iris Plus Graphics, HD Audio with stereo speakers, HP True Vision HD camera, Realtek RTL8821CE 802.11b/g/n/ac. The Python Programming is done on Windows 10 64 bit Operating System platform.

b. Dataset name

ICVL dataset, KTH dataset, SGSITS College (custom dataset), Indore Railway Station (custom dataset), Guwahati Railway Station (custom dataset), New Delhi-Howrah Train (custom dataset), video dataset (custom dataset).

c. Dataset for training

- We have 2500 images in the training section, each approximately 3 MB in size.
- 8 h real-time video

d. Dataset for testing

- 1500 images in the testing section
- 2 h real-time video

3. Proposed work

The work performed here is for the real-time detection of individual actions via video surveillance systems. Human activity regions are
detected using the nonlinear Bayesian filter algorithm. Three CNNs were employed to classify three categories, namely shape, motion history, and combination cues. The classifier predicts output for each region based on the sub action description.

Due to the increase in social gatherings during COVID-19, we proposed Ex-Mask R-CNN to verify human face masks and perform conventional anomaly detection using live images or videos. Fig. 1 depicts the suggested architecture, which includes mask identification and traditional human detection of an anomaly. It also describes the general design of the real-time conventional anomaly detection model.

The classifiers’ output is post-processed to obtain the appropriate results and make final judgments. The proposed architectural diagram is depicted in Fig. 1. The proposed real-time conventional anomaly detection model’s overall structure. Three CNNs are fed appearance-based temporal aspects of Regions of Interest (ROIs) using a motion-detection, human-detection, and multiple-tracking algorithm. The CNNs create predictions using shape, motion history, and combination cues. Each action is subdivided into three sub-action categories by the sub-action descriptor, which provides a comprehensive description of human activity. Fig. 1 depicts the proposed architecture in its entirety, which includes mask identification and human conventional anomaly detection. Fig. 2 is a subset of Fig. 1 in the training section, illustrating the processing of images.

Facial Mask Detection with Instance Segmentation using Extended Mask R-CNN.

- Finding the region of interest by using wrapping instead of pooling.
- Generating region matrix using P, defined algorithm 1.
- Introducing interpolation in finding proposal region.

### 3.1. Sub-action descriptor

The challenge of expressing an activity is not well-defined in terms of a geometric measurement problem (e.g., measurement of an image or camera motion). To provide extensive information on human activities and to help clarify action information, this research proposes a model of an action using a sub-action descriptor. The descriptor has three levels: posture, mobility, and gesture. The connecting line between two sub-actions at various levels denotes their independence from one another. No connection implies an incompatibility between the two sub-actions.

### 3.2. Multi-CNN action classifier

During the training phase, the ROI and sub-action annotations are manually determined for each frame of the training videos, and the ROI is then used to compute three appearance-based temporal features: Binary Difference Image (BDI), Motion History Image (MHI), and Weighted Average Image (WAI).

**BDI: Binary Difference Image.**

The BDI feature accurately captures the actor’s static shape cue in two-dimensional frames, denoted by the variable \( b(x, y, t) \), as defined by Eq. (19).

\[
\begin{align*}
\forall x, y, t, & \quad b(x, y, t) = \\
& \begin{cases}
\text{255} & \quad \text{if } d(x, y, t) - f(x, y, t) > \tau_{th} \\
\text{0} & \quad \text{otherwise}
\end{cases}
\end{align*}
\]

Where the values in the BDI are set to 255 if the difference between the current frame \( f(x, y, t) \) and the background frame \( f(x, y, t_0) \) of the input video is bigger than a threshold \( \tau_{th} \), and \( x \) and \( y \) are indexes in the image domain. BDI is a binary image that indicates the silhouette of the posture.

**MHI: Motion History Image.**

The pixel intensity in a motion history image is a function of the temporal history of motion at that moment. MHI captures the actor’s motion history patterns, indicated by \( h(x, y, t) \), and is defined in Eq. 20–22 using a simple replacement and decay operator.

\[
\begin{align*}
\forall x, y, t, & \quad d(x, y, t) = \\
& \begin{cases}
\text{255} & \quad \text{if } d(x, y, t) - f(x, y, t-1) > \tau_{th} \\
\text{0} & \quad \text{otherwise}
\end{cases}
\end{align*}
\]

\[
\begin{align*}
\forall x, y, t, & \quad h(x, y, t) = \\
& \max(0, h(x, y, t) - \Delta t), \quad \text{otherwise}
\end{align*}
\]

\[
\Delta t = \frac{\tau_{max} - \tau_{min}}{n}
\]
MHI is used to measure locomotion, which includes standing, walking, and running. It is formed in Eq. (20) by subtracting the present frame f(x, y, t) from the prior frame f(x, y, t-1). The MHI at time t is calculated for each frame using the result of the preceding MHI. As a result, this temporal characteristic does not need to be calculated again for the entire collection of frames. In Eq. (22), n denotes the number of frames to account while calculating the action history capacity. The hyper-parameter n plays a key role in establishing the action’s temporal range. Although an MHI with a big n spans a significant amount of action history, it is insensitive to present activities. Similarly, MHI with a small n prioritizes recent acts and overlooks previous ones. As a result, selecting a suitable n can be somewhat challenging.

**WAI: Weighted Average Image.**

Weighted Average Images (WAIs) are used at the gesture level of the sub-action descriptor, which includes inactivity, texting, and smoking. The simplest method to recognize delicate acts (e.g., texting and smoking) would be to look at the actor’s shape or motion history. The disadvantages of this technology include the inability to record extensive information about minor actions and sensitivity to background movement, such as camera shaking. Shape and motion history signals together create a spatial-temporal characteristic for subtle actions. WAI is denoted as s(x, y, t). It is constructed as a linear combination of BDI and MHI, given by Eq. (23):

\[
s(x, y, t) = w_1 b(x, y, t) + w_2 h(x, y, t) + s.t.w_1 + w_2
\]

Even as activities get more intricate, WAI is not fully lost. \( w_2 T \) is an additional hyper parameter.

Three temporal features based on appearance are derived from human behaviors for the BDI, MHI, and WAI CNNs. The first CNN (BDI) accepts input and detects the shape of the actor in the given data. Following that, it utilizes MHI to determine the motion of the actor’s history. The final one, which is based on WAI, records the actor’s motion history and shape. CNNs are used to classify actions, and each CNN is trained consecutively. Selecting the architecture required for CNN presents difficulties, as it is highly dependent on the application for which it will be utilized. We have retained a lightweight CNN architecture that detects real-time human actions to achieve a fast computation time. All CNNs have identical architectures. The number of sub-actions at each descriptor is equal to the number of output layers. Finally, we’ve included a Softmax regression layer.

### 3.3. Post processing

This layer, referred to as post-processing, revises the action classifier’s predictions using many CNNs for each action. As indicated in Fig. 1, the two sub-actions positioned at various levels of the descriptor are independent. No connection suggests that the two sub-actions are incompatible. A classifier that makes predictions using several CNNs will be utilized. We have retained a lightweight CNN architecture that includes inactivity, texting, and smoking. The first CNN (BDI) detects real-time human actions to achieve a fast computation time. All CNNs have identical architectures. The number of sub-actions at each descriptor is equal to the number of output layers. Finally, we’ve included a Softmax regression layer.

### 3.4. Algorithmic design

Two methods were proposed for detecting suspicious actions performed by people wearing masks in video frames. First, we proposed the Extended Mask R-CNN algorithm (Ex-Mask R-CNN) to detect individuals wearing masks. Second, we described a technique known as the Optical-Flow Stacked Difference Image (OFSDI). The OFSDI is used to determine wearing masks. Second, we described a technique known as the Optical-Flow Stacked Difference Image (OFSDI). The OFSDI is used to determine wearing masks. Second, we described a technique known as the Optical-Flow Stacked Difference Image (OFSDI). The OFSDI is used to determine wearing masks. Second, we described a technique known as the Optical-Flow Stacked Difference Image (OFSDI). The OFSDI is used to determine wearing masks. Second, we described a technique known as the Optical-Flow Stacked Difference Image (OFSDI).

The entire model trains Apache MXNet, which significantly reduces the time required to train the Ex-Mask R-CNN model—the dotted outer line in Fig. 2 is Apache MXNet. Original Mask R-CNN (He et al., 2020) is a method that makes advantage of ROI pooling. The disadvantage of ROI pooling is that resolution is lost as it is fed through FC layers, soft-max layers, and so on. Then, ROI aligns were introduced in (He et al., 2020), although ROI Align performs better on datasets with a smaller bounding box and fewer recognized objects. To address this issue, Resnet-152 incorporates ROI warping in place of ROI pooling, or Mask R-CNN incorporates ROI Align, which crops and warps a specific ROI on the feature map to a set dimension. The distinction between ROI warping and ROI aligns is that warping modifies the contour of the feature map; it employs bipolar interpolation to extend or contract the picture to the same dimension as ROI align.

**Algorithm 1: Ex-Mask R-CNN.**

Data: Image \( I \)

Results: Face masked pixels are masked by the algorithm

1. **Initiate** Apache MXNet
2. **Begin** \( I \) and Load dataset
3. **Iterate** over every image of the dataset
4. **For each** image \( I \) in dataset \( D \):
   1. **Extract** feature vector from \( I \)
   2. **Append** into feature matrix \( F \)
5. **Pass** matrix \( F \) to pre-trained neural network
6. **Return** region matrix
7. **Calculate** the interested region (P)
   \[ P = \begin{pmatrix} x_1 - x_2 & x_1 - x_3 & x_1 - x_4 \\ x_2 - x_1 & x_2 - x_3 & x_2 - x_4 \\ x_3 - x_1 & x_3 - x_2 & x_3 - x_4 \\ x_4 - x_1 & x_4 - x_2 & x_4 - x_3 \end{pmatrix} \]
   \[ \text{region matrix-intersection} \]
   \[ \text{P-Union} \]
8. **If** (\( \text{IOU} > 0.5 \))
   \[ \text{LABEL} \rightarrow \text{Mask} \}\) Else\( \text{Skip} \) \)
9. **End** ()

Algorithm 1 explains the steps involved in implementing the proposed Ex-Mask R-CNN. To begin, the algorithms initiate Apache MXNet in step 1. In step 2, the Ex-Mask-RCNN algorithm is started. Resnet-152 extracts feature from image \( I \) and return them to \( F \) in steps 3–5. Following that, in steps 6–8, the region proposal network delivers a list of regions that contain images. These regions are then provided to ROI warping, which returns regions of interest and crops and wraps them. The mask is discovered in step 11. At step 12, the algorithm concludes.

Numerical computation algorithm 1st at 9th steps Calculate the interested region (P).

\[
\begin{align*}
&= \begin{pmatrix} x_1 - x_2 & x_1 - x_3 & x_1 - x_4 \\ x_2 - x_1 & x_2 - x_3 & x_2 - x_4 \\ x_3 - x_1 & x_3 - x_2 & x_3 - x_4 \\ x_4 - x_1 & x_4 - x_2 & x_4 - x_3 \end{pmatrix} \\
&= \begin{pmatrix} y - y_1 & x - x_1 & x - x_2 \\ y - y_1 & x - x_1 & x - x_2 \\ y - y_1 & x - x_1 & x - x_2 \end{pmatrix} \\
&= \begin{pmatrix} y - y_1 & x - x_1 & x - x_2 \\ y - y_1 & x - x_1 & x - x_2 \end{pmatrix}
\end{align*}
\]

Compute interested region (P) because we are interested in the region of the frame where our interested object is. In \( P \), x and y of the locator with the Q, the value of p identify intersection over union, which is unable algorithm to select the region which only has—intersection over union greater than a defined threshold.

Numerical computation algorithm 2, with the steps 5 Bayesian filters take \( x \) feature as an input \( \text{in}(x, t) \approx \sum_{k=1}^{N} \delta(x - x_k) \); where t
denotes the time instance, \( k \) constant whose value range from 1 to \( n \), \( z_t \) is an output feature, and \( w_t \) is weight. Which return matrix as an output that gives region for annotation. This annotation region matrix passes through \( O(x, y, t) = \alpha O(x, y, t-1) + D(x, y, t) \), \( t \) is time, \( \alpha \) is update rate \( 0 < \alpha < 1 \), this returns a matrix that contains feature \( O \), which is motion feature matrix. This helps final decision abnormal and normal motion of action.

3.4.2. Optical-flow stacked difference image (OFSDI)

The Kalman filter is used in the Original Mask R-CNN (He et al., 2020). A nonlinear Bayesian filter has replaced this filter. The Kalman filter has two important limitations:

- It makes the assumption that both the system and observation models are linear, which is not true in many real-world circumstances.
- It makes the assumption that state beliefs are Gaussian distributed.

The proposed appearance-based elements have been incorporated into the OFSDI CNN. Combining local, global, and spatial-temporal information extracts a robust and discriminative quality from infrared action data. OFSDI’s architecture is depicted in Fig. 3. Where FP denotes the future passed in Fig. 3. Algorithm 2 explains the OFSDI approach.

Algorithm 2: Activity Algorithm: OFSDI

\[
\text{Begin} () \{ \\
\text{Load dataset } D；\\n\text{Iterate over every image of the dataset}  \\
\text{for each image } i \text{ in dataset } D: \{
\text{Extract HOG features from } i; \\
\text{Append } F \text{ to features matrix } F; \\
\text{Apply nonlinear Bayesian filter on matrix } F \text{ for filtering,} \\
\text{ } P(x|t) = \sum_{z_k} w^{(k)}(x-z_k); \\
\text{Where } x \text{ is a feature,} \\
\text{Returns region Matrix.} \\
\text{Annotate the regions from the region matrix.} \\
\text{Calculate matrix feature } O \text{ from region matrix} \\
\text{ } O(x, y, t) = \alpha O(x, y, t-1) + D(x, y, t); \text{ } t \text{ is time,} \alpha \text{ is update rate} \ 0 < \alpha < 111.3 \text{ Store } O \text{ into a motion feature matrix} \\
\text{If (motion in motion feature matrix} > 0; \\
\text{ } \{message \rightarrow \text{Abnormal} \} \text{ else (message } - \text{normal} \} ) \\
\text{End}()
\]

Algorithm 2 begins with Step 1. The dataset is loaded in step 2. Following that, iteration over each image in the dataset is performed in step 3. In step 4, the image is processed through a Histogram of Oriented Gradients (HOG) combined with a Support Vector Machine (SVM) to extract the picture’s features \( F \). In step 5, \( F \) is used to refine positions using the nonlinear Bayesian filter, which also provides regions of interest ROI. Stage 6: Annotations are also generated across regions during this step. To detect suspicious human activity, these annotated regions are fed to multi-CNN (MCNN) appeared-based models. Finally, in step 8, we observe both normal and deviant behavior. The algorithm then comes to a halt at step 9. The first table lists all symbols with algorithm 1 and algorithm 2 notations.

4. Experimental results

The recognition rate and processing time of the suggested approach for real-time conventional anomaly detection in surveillance videos are tested in this section. We investigate the systematic estimation of many hyper-parameters associated with appearance-based temporal characteristics. Additionally, the proposed OFSDI of the appearance temporal aspects is provided using a CNN-based technique. The conventional anomaly detection findings are demonstrated using the ICVL dataset, which is the only dataset suited for multiple-individual conventional anomaly identification only the dataset dataset created films were used. We use 8 h of video frames for training and 2 h of video frames for testing our model. Additionally, we implemented the proposed method on the KTH dataset (Schüldt, Laptev, & Caputo, 2004) to compare its performance to several current methods.

We have taken action recognition dataset from different sources in different actions of humans. First, we have taken the two data sets in form of image and video. The taken data sets collected the different sources points. First case from SGSITS College events in Indore, second case from the Indore Railway Station, third case from the Guwahati Railway Station, and in the fourth case from the Indore Railway Station, Indore (Madhya Pradesh), India. We have 2500 images in the training section, each approximately 3 Mb in size, from all four examples described above, and 1500 images in the testing section, which belong to all four cases mentioned before. We employed a video dataset to detect activity; every footage was captured in full HD quality, and 10 h of real-time created films were used. We use 8 h of video frames for training and 2 h of video frames for testing our model. The experimental results indicate that when the OFSDI appearance-based temporal features are combined with a multi-CNN classifier, more accurate detection from surveillance videos is possible. Python Programming Language is used to design 15.6 in HD WLED touch screen (1366×768), 10-finger multi-touch support. The next section contains a detailed description of the dataset employed in the experiment.

4.1. Dataset description

We used two different types of data sets: an image dataset and a video dataset. The image data set is applied for human object detection with mask detection and activity analysis, while the video data set is used for human object detection with mask detection and activity analysis. Additionally, these two types of image data sets correspond to four distinct scenarios. Which are images of data sets obtained in the first case from SGSITS College events in Indore, in the second case from the Indore Railway Station, in the third case from the Guwahati Railway Station, and in the fourth case from the Indore Railway Station. Splendid New Delhi-Howrah Train and video data sets are divided into three categories. The first category includes videos of data sets collected at SGSITS College events in Indore; the second category includes data sets collected at Indore Railway Station, Guwahati Railway Station. This category includes data sets collected at ICVL. The data set is separated into two halves, one for training and another for testing, with an 80:20 training to testing ratio.

4.2. Hardware/software used for implementation

Python Programming Language is used to design 15.6 in HD WLED touch screen (1366×768), 10-finger multi-touch support. 10th Generation Intel Core i7-1065G7 1.3 GHz up to 3.9 GHz. 8 GB DDR4 SDRAM
The system uses a 2666 MHz, 512 GB SSD, No Optical Drive, Intel Iris Plus Graphics, HD Audio with stereo speakers, HP True Vision HD camera, Realtek RTL8821CE 802.11b/g/n/ac. The Python Programming is done on Windows 10 64 bit Operating System platform. The following python libraries were used during implementation: NumPy, Pandas, Matplotlib, SciPy.

**Evaluation Metrics.**
At video-based video-AP, we employ average precision to quantify results. Video-AP provides an informative measurement for action detection in video-based evaluation. The Video-AP; in this case, the detection is correct if it meets the frame-AP conditions for spatial domain and frame intersection with the ground truth. If the frames correctly predicted for a single action exceed the temporal domain (τ).

Additionally, the Mean Average Precision (mAP) for all action categories at video-based measurement is used to evaluate the presented approach, as the number of activities visible in a single video is limited, and the distribution of instances within each category is significantly unbalanced on the test and validation sets. In all approaches and throughout experiments, an intersection-over-union threshold of σ = 0.5 and an intersection-over-frames threshold of τ = 0.5 were used.

### 4.3. Results and discussions

This section evaluates the proposed algorithm with mask detection using ICVL and human activity analysis and a video dataset for object detection with mask detection and human activity analysis. Fig. 4 illustrates scenes captured by two distinct cameras at the Indore Railway station. The figure represents a person wearing a mask while engaging in routine tasks. Whereas, Fig. 5 demonstrates that the person is not wearing a mask, as determined by the suggested Ex-Mask R-CNN algorithm and the proposed OFSDI method. Figs. 6 and 7 illustrate human activity associated with face mask detection.

#### 4.3.1. Results of Ex-Mask R-CNN

The Ex-Mask R-CNN results are reported in this part on a series of test images using Resnet152-FPN running at 6 frames per second with 36.3 masks AP. Figs. 8-10 illustrates object detection proposed Ex-Mask R-CNN determines whether or not the object in the frame is a human face.

Using the ICVL dataset without a mask, Fig. 8 illustrates the performance of ResNeXt-152-FPN results from the suggested Ex-Mask R-CNN method. The findings of the Railway Station datasets with the human face mask
are shown in Fig. 9. Fig. 10 depicts the SGSITS college dataset’s results with a human face mask. Figs. 11-13 illustrates whether or not a mask covers a human face, and the suggested Ex-Mask R-CNN detects an object as a human face covered by a mask.

The performance of ResNet-152-FPN is shown in Fig. 11, along with the results of the suggested Ex-Mask R-CNN method on the ICVL dataset without a mask. The findings of the Railway Station datasets with a human face mask are shown in Fig. 12. The results of the SGSITS college dataset with a human face mask are shown in Fig. 13.

The graph above illustrates the performance of Mask-RCNN when several ResNet architectures are used. Additionally, it demonstrates that the ResNet 152 architecture utilized in our proposed model outperforms all other ResNet architectures in terms of evaluating average precision matrices.

4.3.2. Action detection

For BDI-CNN, MHI-CNN, WAI-CNN, and proposed OFSDI, Video-AP is reported. OFSDI outperformed AP measures by a large margin, demonstrating the importance of the combined cues for the job of action recognition. Table 2, Table 3, and Table 4 compares the proposed method to previously published methods for action recognition across all datasets.

On the ICVL dataset, Table 2 summarizes the suggested OFSDI model’s conventional anomaly detection performance. Here, normal refers to the individual performing a normal activity, whereas abnormal refers to performing an abnormal activity or behaving abnormally. The metric used to evaluate the traditional anomaly detection in the image frame is average precision.

On the Railway Station dataset, Table 3 summarizes the performance of the suggested named OFSDI model in terms of traditional anomaly detection. This collection contains images of individuals in various locations within the Railway Station, including inside the train, on the platform, and near the platform. The average precision metric is employed in this case. It performs an analysis of the image frame’s traditional anomaly detection.

Table 4 summarizes the proposed OFSDI model’s conventional anomaly detection performance on the SGSITS College dataset. This dataset is comprised of images collected at various locations within the SGSITS campus, including the entrance, the greenery, and so on. The standard deviation of the average metric precision is used to evaluate the conventional anomaly detection in an image frame.

5. Conclusion

This paper described a novel real-world surveillance video dataset and a method for real-time anomaly detection in video surveillance systems. In the future, the datasets employed in the experimental evaluation will drive research on multiple conventional anomaly detection. According to major implementations of the suggested approach, the sub-action descriptor provides comprehensive information on human actions. It reduces misclassifications caused by a greater number of activities composed of several distinct sub-actions at various levels. Additionally, our suggested approach localizes and recognizes the actions of a large number of individuals with a low computational
Table 2
Results of the action recognition of IVCL dataset.

| Video-AP (%) | Normal | Abnormal | mAP |
|--------------|--------|----------|-----|
| Novel Spatiotemporal Network | 56     | 51       | 53.5|
| NEUCOM21     | 59     | 52       | 55.5|
| Improved Robust Video Saliency Detection based on Long-term Spatial-temporal Information | 69     | 47       | 58  |
| BDI-CNN      | 77.8   | 53.9     | 65.8|
| MHI-CNN      | 70.6   | 64.3     | 67.4|
| WAI-CNN      | 82.9   | 77.1     | 80  |
| OFSDI        | 80.1   | 78.4     | 79.3|

Table 3
Results of the action recognition of Railway Station dataset.

| Video-AP (%) | Normal | Abnormal | mAP |
|--------------|--------|----------|-----|
| Novel Spatiotemporal Network | 68     | 72       | 70  |
| NEUCOM21     | 59     | 71       | 65  |
| Improved Robust Video Saliency Detection based on Long-term Spatial-temporal Information | 69     | 76       | 72.5|
| BDI-CNN      | 77.8   | 53.9     | 65.8|
| MHI-CNN      | 70.6   | 64.3     | 67.4|
| WAI-CNN      | 82.9   | 77.1     | 80  |
| OFSDI        | 80.1   | 78.4     | 79.3|

Table 4
Results of the action recognition of the College dataset.

| Video-AP (%) | Normal | Abnormal | mAP |
|--------------|--------|----------|-----|
| Novel Spatiotemporal Network | 68     | 61       | 64.5|
| NEUCOM21     | 71     | 52       | 61.5|
| Improved Robust Video Saliency Detection based on Long-term Spatial-temporal Information | 69     | 51       | 60  |
| BDI-CNN      | 75.8   | 51.9     | 63.85|
| MHI-CNN      | 69.6   | 61.3     | 65.4|
| WAI-CNN      | 79.9   | 74.1     | 77  |
| OFSDI        | 82.1   | 78.4     | 80.25|

Fig. 12. Railway Station dataset with the human face mask.

Fig. 13. SGSITS college dataset with a human face with a mask.

cost and high accuracy. For face mask recognition, the proposed Ex-Mask R-CNN outperforms other Mask R-CNNs. In the future, we can adapt our proposed approach for big data platforms to various datasets.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgement

The authors like to express their gratitude to the anonymous referees for their constructive comments that contributed to the paper’s improvement in quality.

References

Ali, S., Basharat, A., & Shah, M. (2007). Chaotic invariants for human action recognition. In Proceedings of the IEEE International Conference on Computer Vision. https://doi.org/10.1109/ICCV.2007.4409046
Brox, T., Bruhn, A., Papenberg, N., & Weickert, J. (2004). High accuracy optical flow estimation based on a theory for warping. Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 3024, 25–36. https://doi.org/10.1007/978-3-540-24673-2_3
Chavda, A., Dsouza, J., Badgjar, S., & Damani, A. (2020). Multi-Stage CNN Architecture for Face Mask Detection. http://arxiv.org/abs/2009.07627.
Dalal, N., & Triggs, B. (2005). Histograms of oriented gradients for human detection. Proceedings – 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2005, 1, 866–893. https://doi.org/10.1109/CVPR.2005.177.
Dalal, N., Triggs, B., & Schmid, C. (2006). Human detection using oriented histograms of flow and appearance. In Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics). https://doi.org/10.1007/11744047_33
Dhiman, C., & Vidwakarma, D. K. (2019). A review of state-of-the-art techniques for abnormal human activity recognition. Engineering Applications of Artificial Intelligence, 77, 21–45. https://doi.org/10.1016/j.engappai.2018.08.014
Dollár, P., Rabaud, V., Cottrell, G., & Belonge, S. (2005). Behavior recognition via sparse spatio-temporal features. Proceedings – 2nd Joint IEEE International Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance. VS-PETS. 2005, 65–72. https://doi.org/10.1109/VSPETS.2005.1570899
Elsayed, O. A., Mohamed Marzouky, N. A., Atef, E., & Salem, M. A. M. (2019). Abnormal Conventional anomaly detection in video surveillance. In Proceedings – 2019 IEEE 9th International Conference on Intelligent Computing and Information Systems. ICICIS. https://doi.org/10.1109/ICICIS46948.2019.9014712.
Fathi, A., & Mori, G. (2008). Action recognition by learning mid-level motion features. In Proceedings of the IEEE International Conference on Computer Vision. https://doi.org/10.1109/ICCV.2007.4409046
Feng, S., Shen, C., Xia, N., Song, W., Fan, M., & Cowling, B. J. (2020). Rational use of face masks in the COVID-19 pandemic. In The Lancet Respiratory Medicine (Vol. 8, Issue 10.1109/TPAMI.2018.2844175
He, K., Gkioxari, G., Dollár, P., & Girshick, R. (2020). Mask R-CNN. IEEE Transactions on Pattern Analysis and Machine Intelligence, 42(2), 386–397. https://doi.org/10.1109/TPAMI.2018.2844175
Jin, C.-B., et al. (2017). Real-Time Action Detection in Video Surveillance using Sub-Action Descriptor with Multi-CNN. ArXiv abs/1710.03283, n. pag.
Loey, M., Manogaran, G., Taha, M. H. N., & Khalifa, N. E. M. (2021). A hybrid deep transfer learning model with machine learning methods for face mask detection in the era of the COVID-19 pandemic. Measurement: Journal of the International Measurement Confederation, 167, Article 108288. https://doi.org/10.1016/j.measurement.2020.108288
