ORIGINAL ARTICLE

An Event Detection Mechanism with Deep Feature Extraction and Optimal Loss Function Based XGBoost Classifier

B Manjula[1*], P. Venkateshwarlu[2]

[1] *Department of Computer Science, University College, Kakatiya University, Telangana, India*

[2] *Department of Computer Science, Vaageswari College of Engineering, Telangana, India*

*Corresponding Author: B Manjula
Email: purumalavenkateshwarlu@gmail.com

Received: 02 December 2022 / Accepted: 02 April 2023

Abstract

Purpose: Interoceptions are a combination of sensation, integration, and interpretation of internal bodily signals. Interoceptions are bidirectionally related to the human being mental and physiological health, and well-being. Sleep and different interoceptive modalities are proven to share common relations.

Heartbeat Evoked Potential (HEP) is known as a robust readout to interoceptive processes. In this study, we focused on the relation between HEP modulations and sleep-related disorders.

Materials and Methods: We investigated four different sleep-related disorders, including insomnia, rapid eye movement behavior disorder, periodic limb movements and nocturnal frontal lobe epilepsy, and provided HEP signals of multiple Electroencephalogram (EEG) channels over the right hemisphere to compare these disorders with the control group. Here, we investigated and compared the results of 35 subjects, including seven subjects for the control group and seven subjects for each of above-mentioned sleep disorders.

Results: By comparing HEP responses of the control group with sleep-related patients' groups, statistically significant HEP differences were detected over right hemisphere EEG channels, including FP2, F4, C4, P4, and O2 channels. These significant differences were also observed over the grand average HEP amplitude activity of channels over the right hemisphere in the sleep-related disorders as well.

Conclusion: Our results between the control group and groups of patients suffering from sleep-related disorders demonstrated that during different stages of sleep, HEPs show significant differences over multiple right hemisphere EEG channels. Interestingly, by comparing different sleep disorders with each other, we observed that each of these HEP differences' patterns over specific channels and during certain sleep stages bear considerable resemblances to each other.

Keywords: Support Vector Machine; Random Forest Classifier; Deep Stacked Auto-Encoder; XGBoost Classifier Extreme Gradient Boost Classifier; Classification; Event Detection.

DOI: https://doi.org/10.18502/fbt.v11i3.15879

1. Introduction

In the contemporary era, event recognition has attained widespread attention as it seems to be applicable in different category of application from personal events to collective events etc. Other side, to follow the immense success rate in event classification, detection and recognition of objects, Deep Learning (DL) emerges to exhibit high performance in event detection tasks. Hence, a larger portion of researchers had a dependency on DL architecture for event detection. It establishes how the deep features had changed the framework of event recognition [1].

Unlike the various elementary visual concepts, the image complex events were high abstraction levels of richer content and long temporal spans having high dramatic variations. The web videos that describe the events were commonly large in image size, high label sparsity and noisy content apart from images utilized for the content-detection study. Hence, in such complex detection of events, there are various challenges posed and that imposes the necessity for efficient algorithms. These algorithms must enable in building of an MED (Multi-Media Event Detection) model, that is utilized practically in complex-event detection [2].

A larger event-related videos/images collection was required to fulfilling training-requirement indulged in deep architecture. In this scenario, large training data is considerably required in comparison to conventional techniques [3]. Even though different benchmark datasets were available, none of the datasets were larger enough to get utilized in deep architecture training from the point of scratch. By focussing on these challenges that are related to training data, at the forefront two kinds of approaches exist inclusive of synthetic data-generation and transfer-learning. In this transfer-learning, an existing pre-trained model was subjected to fine-tuning of features, upon event-related images [2]. Concurrently, in synthetic data augmentation, data training sets were populated through the generation of synthetic training set images by various techniques including rotation and cropping [4]. Other side, another challenge is handling the limitation of holding memory, different tricks through small batch-size of image, distributing the model upon different machines, or decreasing the size of the model. To handle these dataset complexities having more spatial features [5], fine-tuning process upon event-associated images could rectify the challenges related to the requirements of training data and enhance the performance outcomes. In the fine-tuning process, the conventional model was pre-trained upon a larger dataset, including placed dataset and ImageNet which were tuned upon event-related images through commencing learning phase. This fine-tuning stage could be initiated through varying numbers and names of output feeds of the last layers [6].

The existing researchers used a deep model to learn the features wherein the parameters learned upon generic data sets retrieve features from event-related image collection. The pre-trained model on ImageNet extracts object-level information wherein scene-level features were extracted by pre-training on placed dataset [7]. Although the feature choice relies on application nature, deep learned feature proves to be efficient in comparison to hand-crafted visual features in various domains like natural disaster images in social media, image retrieval, and re-identification of persons. In these models, features are extracted generally from last-fully connected-layer and then VGGNet model is used for all ResNet configurations through eliminating top-layer for feature classification. But, this features could be extracted from a single model layer [8]. The feature vector's length differs with various network architectures. These features were utilized in training classifiers such as RF, Softmax, and SVM classifiers aiding in the prediction of events [9]. Hence, in order to avoid the classification issue, misclassifying the feature, imbalanced dataset issue, and data loss due to deep layer learning of features, the study exposes to bring out effective Deep learning based event detection method using VGG-16 and ResNet-50 models for feature extraction, deep stacked auto-encoder for feature fusion phase and optimal loss XGBoost for classification [10].

1.1. Problem Identification

It becomes impractical to assess the data manually for extracting significant or newsworthy content, from the image set, due to its dynamic nature and huge volume. Hence, to use social media data efficiently, the necessity of accurate and automated event detection techniques is highly critical. To be the extension towards Embed2Detect, high advanced image embedding technique could be employed. But the same event detection method, without leaving deep features must consider the associated complexities and learning time, to maintain the method's efficacy [11]. In this scenario, it is highly suitable to utilize embedding models, or deep neural networks to handle this sort of imbalance data issue. Even though the data imbalance is dealt with event detection, such methods must cope with covering all the significant features during the training phase as deeper layers with convolutional filters are indulged in architecture. For instance, the Skip-gram method such as BERT-model was more advanced than normal event classification and detection approaches. This kind of reconstruction issue in framing out the original pixel representation from input data is another challenge that ought to get handled in such an approach [12].

Fusing the features of image through DL could be efficient. The notion behind performing feature fusion involves integrating several representations or features of the image to attain robust and comprehensive image representation. Through the fusion of image features with DL, the model could learn in weighting the significance of individual features and then integrate them ideally for enhancing the complete performance of the classifier. For instance, in fusing the deep features from varied images like depth features and RGB, the model could learn to utilize harmonizing information from individual sources for making suitable predictions.

In this strategy, DL is employed for feature extraction from images that could be considered as a data representation form. Later, these features are fed into a proposed classifier that permits the strengths of ML and DL to be complementarily leveraged.

DL is especially efficient in learning complex and hierarchical data representations like images. It has an innate ability to work better than conventional hand-crafted features in several computer vision-based tasks. Nevertheless, DL models could be difficult for training and computationally intensive for big datasets.

On the contrary, ML algorithms are generally easier and faster to train. It could also effectively deal with huge datasets. Through the use of ML algorithm to perform classification, it turns probable to learn the models, while being capable of scaling with large datasets. Hence, integrating DL-based feature extraction with ML-based classification could be an efficient approach to solving computer vision issues (handling imbalanced datasets and loss and vanishing gradient problems for improvising the learning rate) as it permits the strengths corresponding to both methodologies to be used complementarily.

1.2. Objectives of the study

In order to address the discussed complexities above, this study focuses to classify and determine events with better accuracy; specifically, the paper adds the following contributions,

- To explicate features extraction mechanism using VGG-16 and ResNet-50 model to prevent vanishing gradient problem as deeper layer-wise learning, prone to have data loss.
- To accomplish feature fusion using deep stacked auto-encoder to solve issues on insufficient data and reconstruction error.
- To classify the event features using optimal loss function with XGBoost classifier to eliminate over-fitting issues, label imbalanced datasets, and mislabelling of features.
- To evaluate the proposed framework by assessing its performance with regard to metrics for confirming its efficacy in event detection.

1.3. Paper Organization

The organization of the paper is stated in the following sections. Section I elucidates the fundamental aspects of event detection and the problems identified. Then, section II reviews the traditional research in this domain and their underlying machine learning, deep learning, and other metaheuristic approaches. The proposed system is comprehensively provided by enumerating all the introduced algorithms and dataset descriptions in section III. Results are depicted in section IV which is obtained through the analysis of the proposed system; similarly, comparative analysis was illustrated in the same section to depict the efficacy of the proposed system than a conventional system. Finally, the overall summary of the system is concluded in section V.

1.4. Review of Literature

The below section enumerated a review analysis of existing researchers, discussing the different approaches of event detection and its application usage.

1.4.1. Existing Methods on Event Classification and Detection Model

To consider the significance of event detection within the social media context, various techniques were proposed by past researchers with associating distinct characteristics and techniques such as the clustering approach, rule mining method social aspects, graph theory, Machine Learning (ML), and Deep-Learning (DL) methods. The method had attained substantial development in the prediction and classification of events from clubbed images. However, certain researchers leverage the DL technique for real-disaster management and detection. This may be primarily because of constrained annotated available data in that domain. The previous researcher gathers information from social media or the web and performs annotation manually [13]. The image variability in those datasets might not seem sufficient in creating the model robustness, utilized in various real-world circumstances. For instance, different flood images are obtained in day time, many users do posting of clear images without the presence of noise. The trained model on those data could not determine the disaster even easily from those noisy real-world images. One such (CapsNet) Capsule neural network was introduced recently in the field of image processing with an intention to rectify the CNN known limitation, particularly in robustness towards affine transformation like orientation, size of the image, and overlapped image detection. This aspect motivates researcher in employing CapsNets in handling polyphonic event detection wherein different events prevail simultaneously [14]. Particularly, capsule units exploit the representation of a group of distinctive attributes of images for single event are proposed. The capsule units were connected using dynamic-routing which encourages learning the whole feature complex relationship layer by layer and enhances the polyphonic context performance [15].

Deep learning applications in the conventional techniques, including DeepVoG were premised on the outcomes on PC equipped with GPU and hence necessitate the larger memory space and indulge higher complexity of computation capacity due to the larger parameter count utilized in ConvNet. The real time gazes out prediction and detection of events; hence it becomes highly tedious in carrying out smaller devices with constrained computing resources. Moreover in DeepVoG, the first instance the ConvNet post-processing outcomes after the computing direction necessitates the algorithmic usage to detect the eye movement events including fixed gaze, saccades, and blinking [16]. Hence to propose an effective image-processing technique for event detection and gaze inference, that is capable of using compact devices. The method utilized end-to-end processing NN, in processing gaze estimation outputs and event detection in eye movement on the basis of input movement video [17]. Similarly, another research outlines a multi-model event-detection method through the integration of RGB, audio models, and optical flows using robust MobileNet and ResNet50 architectures employed upon sagemaker. The outcomes of the study described that through parallel development of the model, the multi-model event detection enhanced the outcomes in performance in 25-class event determination in sports-stream [18].

The method to detect the soccer match events utilizes two CNN and Variational Auto-Encoder (VAE). Such research concentrates to resolve the issues of no highlighted image frames that had a chance to get classified incorrectly as any events and to deal with the issues of similarity among yellow and red card frames [19, 20]. Owing to this method, the hybrid algorithm using recurrent neural model and CNN in picking the phases from achieved continuous waveforms in two different steps. At first, the eight-layered CNN was trained in detecting the earthquake events ranging from thirty-second long three-component seismograms. These events determined the seismograms were sent towards two-layer bi-directional RNN model in picking up S-arrival and P-arrival times [21].

Likewise, the transferred Deep CNN approach based anomaly-detection method and its architecture for hyper-spectral imagery dataset through labelled sample pixel pairs are implemented. This model represents different and same training classes and testing phases. The different classes in the testing and training phases were exhibited on the basis of variation between voting mechanisms and neighboring pixels in detecting anomalies. The approach necessitates the tuning of different parameters including learning rate and window size and RNN-based DP employed by Lyu. This method is utilized in the detection of changes in land coverage within hyper-spectral multi-temporal images. The extension of the work is performed by doing a more comparative analysis of DL-based methods with different Non-CNN methods. The comparative assessments revealed that the approach robustness incorporates the environmental variables for other extreme detection of events [22].

Similarly, the attention over the multiple images was focused on multi-modal event detection. This sort of technique was highly reasonable with multiple images and shorter text for tweets. To this end [23], the novel MIFN-Multi-image focusing-network is elaborated for connecting text context, having visual parameter within multiple images. This MIFN approach comprises of feature-extractor, event classifier, and multi-focal network [24]. This multi-focal network applies the focal attention over different images, and it fuses out the related features to be a multi-modal representation. The classification of the events finally estimates the social events on the basis of multi-model representations [25]. The effectiveness of the proposed method was evaluated through extensive experiments on commonly utilized disaster data-set.

Similarly, another event detection model, represented as a background agnostic model learns out the features from model training videos comprising of normal events. This framework consists of object detectors, motion auto-encoders, groups of classifiers, and appearance scenes. This framework focused on only the detection of objects such that it could be employed in various scenes, wherein the normal events were identically defined over the scenes. The single major variation factor is background. This turns out the method into background-agnostic, since it strictly depends upon objects causing anomalies, however, not on the background [26]. In order to rectify the abnormal data lacking in the training phase, the adversarial learning strategy for auto-encoders is proposed. In this method, the scene-agnostic groups of out-of-domain pseudo-abnormal instances are reconstructed correctly by those auto-encoders, before gradient-ascent is applied to those pseudo-abnormal instances. The pseudo-abnormal examples were further utilized in serving as abnormal instances while motion-based and appearance-based binary classifier training occurred. These classifiers discriminate the reconstructions, abnormal, and normal latent features. Most of the CNN architecture decreases the spatial dimension to a low count, while it increases the kernels count utilized. In the classification of images, the method making out the sense as absolute object's position in the image does not matter; hence, the spatial dimension reduction would lead to translation invariance, beneficial in the classification of images [27].

The universal and simple approach in online events detection denoted by abrupt bursts on the basis of simple gradient-based non-linear signal transformation in longer-term observational data series, to be the product of derivative and signal was proposed in another research [28]. However, in the form of video clips, the static features of the image frame extracted by CNN and the features were fused by RNN, for representing the video in common length to select those features from three dimensional images. Niet, in his research, enhanced such CNN model that utilized low-level CNN features to be the content and utilized RNN model to retrieve deep features. On such extracted features, the nearest-neighbor classifier utilized to exhibit model's viability. The better hyper parameters setting yields better outcomes specifically by using minimum input-frames in each video [29].

On the basis of the pre-trained detector concept obtained from other external sources, another learns the semantic correlation from vocabulary and it emphasizes common concepts for zero shot-event detection. The weight parameter was computed through AUC-area under the curve. The weights that are assigned from the pre-trained model were integrated into the confidence score vector to effectively characterize event correlation statistically [30]. Likewise, transfer learning engages to copy the learned weights, trained on base-model towards the target model. This capability enhances the accuracies of the model, decreases the training time of the model, and decreases required volume of labeled data. In order to differentiate how the models were trained without involving transfer learning [31].

1.4.2. Research Gaps

The long-term goal is to enable the system for understanding complex events in different unconstrained environments through limited resources. The concrete problem is to detect efficiently those complex events from videos or set of images.

However, the source of videos or sets of images represents real-world complexity because of the larger diversity in language, production qualities, style, content etc. In accordance with Cisco, the content of video would acquire approximately 82 percent of the total traffic of internet by 2020 [32]. The accuracy of event classification was lower for rare classes of events that represent a clear CNN model limitation. it is common to determine cases having low confidence measure of model and thereby making sure that could be reviewed by experts. Norouzzadeh addressed this issue by assigning high weights on rare event classes in the training phase of the model; however, the models were not capable of enhancing systematically the accuracies in feature classification. Even in Deep neural networks, as features are learned in deep layers, the data loss occurs gradually. Similarly, class imbalance is not pointed out in event classification during modeling of neural networks [33].

Once the model is trained, the images were not confronted with training models, like image with new class, new locations, and new camera angles must be carefully monitored. In those situations, researchers must assess the performance of the model carefully and must retrain the model if it is required. The researcher must give keen attention to equate the CNN output towards real probabilities. The researchers explicated that DNN tends to get confident in event predictions [34].

2. Research Method

The study intends to propose an Event Detection Mechanism using VGG-16 and ReNet-50 Feature Extraction with Optimal Loss Functionality of the XGBoost Classifier. Although conventional works have performed better, they need further enhancement in accordance with handling imbalanced datasets and loss and vanishing gradient problems for improving the learning rate. To achieve this, the present study is proposed with the overall flow as described in Figure 1 with Feature Classification flow in Figure 2.

As shown in Figure 1, the dataset is loaded into the model, and the input parameters of the images are subjected to pre-processing phase, wherein resizing of the image occurs. The images with fixed size are moved on by pre-processing phase. The image features like pixel by pixel were obtained through VGG-16 design, such that the depth of convolutional network was understood by VGG-16. The model segregates the input features layer by layer and learns the feature vectors. However, it faces the gradient problem that affects the learning rate of the layer-wise extraction of images. As soon as the layer-wise information is learned by more segregation of layers, the learning rate of phase gets decreased, prompting incorrect learning. Hence, this learning rate could be enhanced by applying ResNet-50 model and make the CNN traverse deeper and deeper. The ResNet-50 has introduced residual connection among layers, such that the layer's output is convolutional of input and their input feed. Hence, the ResNet-50 utilized batch normalization that had been integrated into this VGG-16

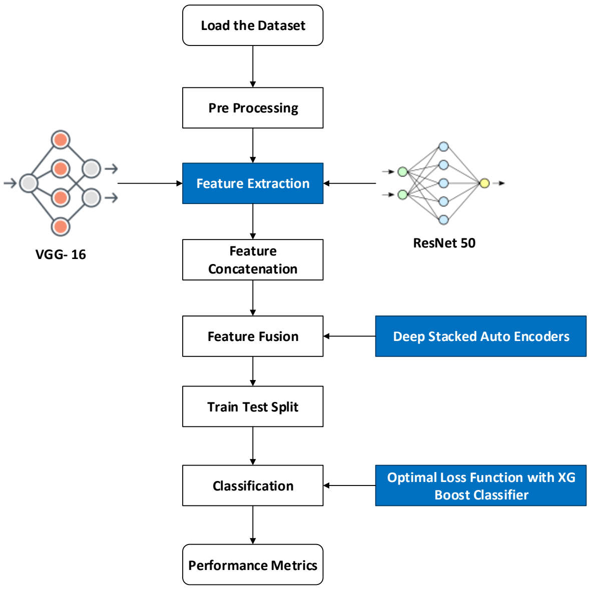


Figure 1. Proposed flow

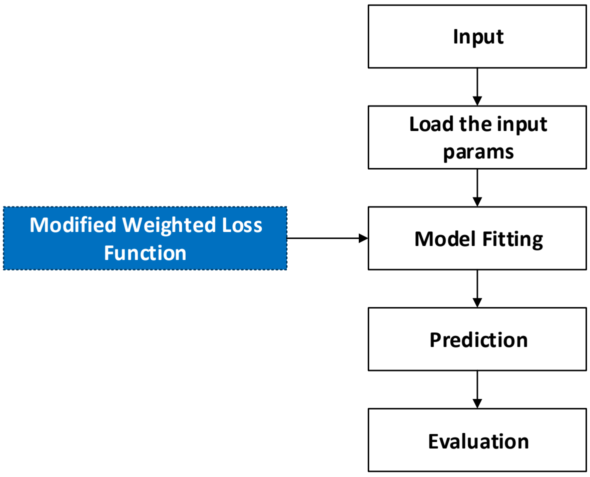


Figure 2. Feature Classification flow - Optimal Loss Function with XG Boost Algorithm

model. The Resent could categorize image features into more categories of a layer such as function parties category, graduation images, dancing images etc. avoiding the vanishing gradient problems.

More deep layers pave to more accurate results, employed using the ResNet model, with preventing vanishing gradient issue. The large-scale image features extraction with deeper layers are dealt with and concatenated in the ResNet 50 and VGG-16 models. As the features are learned layer by layer, data loss could also be rectified. The features concatenated were fused in the feature fusion phase through the deep stacked auto-encoder technique. The auto-encoder was utilized in the dimensions reduction of data. It is an unsupervised Artificial Neural Network (ANN) for automatic extraction of features with reduced features. The critical features of data were picked out by the encoder and original data recreation is accomplished through critical components. However, the characteristics of data features were not retained having some data loss. At this perspective, the single auto-encoder may be unable to decrease the input features' dimensionality. This reduction in dimensions could not be performed if there are complex feature relationships, like there may be chances to miss the outcomes of some hidden layers, and some layers may not be recognized by this auto encoder. This sort of issue was dealt with by a deep stacked auto-coder, wherein multiple encoders have been stacked on top of one another. Both the input and output of the first auto-encoder layer were fed as the input to the next 2nd auto-encoder. Insufficient data condition could be handled, hence, solved by this deep stacked auto-encoder model in the feature fusion phase incorporating the data features. Similarly, the feature classification, undertaken by the XGBoost classifier enhances the performance outcomes, and the overall process is shown in Figure 2.

However, while classifying the encoded feature vectors, issues in binary or multiple classification prevails, which interprets the data ‘0’ as ‘1’, while in deeper training of features. This binary classification problem, rectified by optimal loss functionality, Modified extreme Gradient boosting algorithm with weighing distance for data features are implemented with loss function to improve accuracy and address the issues of binary classification and complexity of the model. To address the imbalanced issue of labels in the binary regime, weighted XGBoost classifier with a cross-entropy loss function. This function is a cost-sensitive technique to train and classify the imbalanced data and handle the misclassification of information. Then, the trained parameters are fitted into the model as classified features, segregating the events. These classification outcomes move on to the prediction of events identification. The evaluation of the model is held through performance assessment.

As exposed in Figure 2, the input parameters are loaded. The model fitting is performed based on the Modified Weighted Loss Function based on which the prediction is undertaken. Lastly, evaluation is undertaken to determine the effectiveness of the proposed model.

2.1. Dataset Description

2.1.1. Social Event Image Dataset (SocEID)

The dataset consists of nearly 525000 images, arranged into various categories of social events, chosen from most shared ones obtained from the social networks. In order to make balanced dataset, an equal count of images was gathered in each event class (around 35000) using the corresponding API from Flickr. In this collection of images, the best point is in covering each aspect of social events through gathering images for the same kind of events. These sets of events have diversified contents with respect to colors, viewpoints, group pictures versus outdoor and single portraits versus indoor images wherein higher variability of information could be explored effectively in ensuring a better rate of performance in the classification of events. For instance, in weddings, sports, and graduation event classes, pictures at celebration time, single-person pictures, and group pictures were taken. In similar to this, in mountain trip classes and in ski-holiday classes, the dataset will cover pictures of images in white and bare mountains and green mountains (Table 1). One more significant characteristic of the dataset is its diversity within culture (Table 2).

2.1.2. Pre-Processing

In the pre-processing phase, the image features of the pixels undergoes pre-processing phase that removes any redundant values and missed out values in the array representation of image pixels.

Table 1. Sample images from the dataset

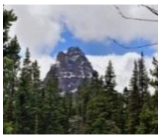
Sea holiday					
Mountain trip					
Exhibition					

Table 2. Training and Testing images count of Dataset and Link

Attribute	Particulars
Name	Events Dataset
Link	http://loki.disi.unitn.it/~used/
No. of Training Images	12000
No. of Testing Images	2400
No. of Events	8

2.1.3. Feature Extraction–VGGNet and ResNet Neural Network Model for Vanishing Gradient Problem and Improvise Learning Rate

After pre-processing phase, the VGGNet-16 network model extracts the features with deeper layers and produces the features irrespective of the depth of the convolutional layer. Generally, as the neural network goes deeper, the pixel distribution and image compression get lower. In this scenario, once VGG-16 neural network is used, the depth in extracting more features, through many layers, does not affect the accuracy of the data. VGG-16 is a kind of CNN model that assesses the network by increasing depth through architecture having 3 × 3 smaller convolutional-layer to improve the feature extraction phase. However, when going to deeper layers, segregating the accuracy of output retrieved from given input paves the way for the vanishing gradient problems. These vanishing gradient problems could be solved by incorporating the ResNet-50 layer. In this network, the 3x3 convolution kernel is utilized in entire layers. The parameters in CONV layers were reduced and it enhances the time of extraction training. The utilization of the ResNet-50 layer is introduced with residual connection among layers. Hence, no residual loss of data occurs during the extraction of the features. This is integrated into VGG-16 to extract the data from deeper layers with good accuracy by avoiding gradient problems (Figure 3). The ResNet-50 could classify the images into 1000 categories of objects, including function parties, graduation dancing images, etc (Figure 4).

2.1.3.1. Architecture Diagram

After pre-processing phase, the VGG-16, a CNN model with sixteen deep layers is used with ResNet-50 for feature extraction, and it classifies the images into different categories. In this VGG-16, there are nearly thirteen convolutional layers, three dense layers and five max-pooling layers. These layers were summed up to twenty-one layers; however, it possesses only 16 weight layers. Each and every feature from pre-processed step was learned layer by layer deeper in this incorporation of VGG-16 and ResNet-50 layers. VGG-16 model obtains input tensor size to be 224, 244 having three RGB channels. Activation function values are added to each feature in each layer of the VGG-16 and ResNet-50 models. In VGG-16 architecture, rather than having larger hyperparameters count, the model focused to have convolutional layers having 3×3 filters with stride-1 or stride-2. The max-pool layers and convolutional layers

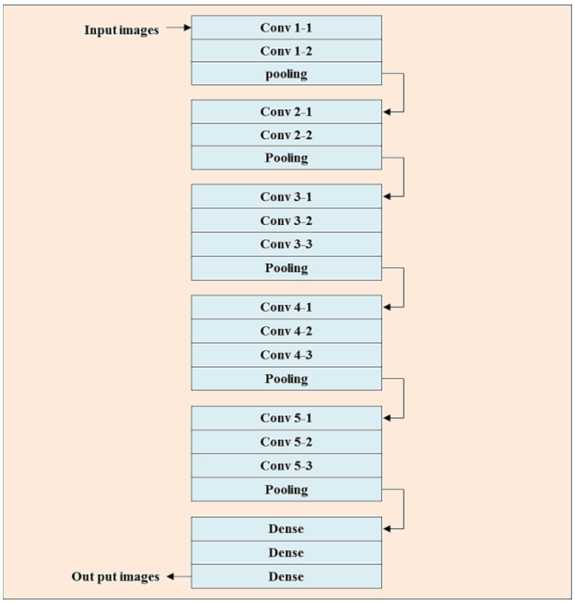


Figure 3. Architecture Design of the VGG-16 model

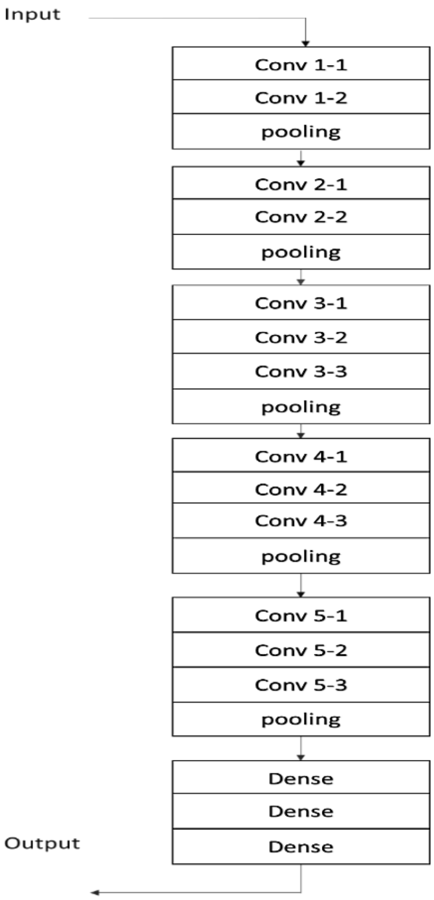


Figure 4. Architecture Design of the ResNet-50 model

were arranged consistently in the entire architecture. Conv-1 Layer consists of 64 filters, 128 filters in Conv-2, 256 filters in Conv-3 layer, and nearly 512 filters in Conv-4 and Conv-5 layers. Three Fully Connected (FC) layers that proceed the stack of convolutional layer consist of 1000 channels each for a single class. The soft-max layer is the final layer. To avoid the vanishing gradient issue, the ResNet-50 neural network is integrated with the VGG-16 design. The ResNet-50 design comprises five stages; every stage comprises identity and convolutional block.

Both ResNet-50 and VGG-16 models push out a depth of 16-19 weight layers to exhibit deep single-layer output, to bring out a better rate of accuracy. Hence, the trainable parameters are learned in this way. Every convolutional block in ResNet consists of three convolutional layers and every identity block possesses three convolutional layers. ResNet consists of nearly twenty-three million trainable parameters. By passing each layer of the VGG-16 and ResNet-50 models, deeper layers are extracted without data loss.

2.1.4. Deep Stacked Auto-Encoder for Feature Fusion

Auto-encoder is a category of an unsupervised learning structure, owning three layers, such as input layer, hidden-layer, and then output layer. The auto-encoder training process comprises two parts, including the encoder part and the decoder part. Encoder was utilized to map out the input data to hidden representation and this decoder represents to reconstruct input data from hidden representation. With unlabelled input-dataset (Equation 1).

$$o_n=c(b_1x_n+j_1) \qquad (1)$$

Where this c represents the encoding function, and the encoder weight matrix is denoted by b_1 and the bias vector is pointed by j_1. The process of decoding is explained by the equation below (Equation 2).

$$\dot{X}_n=s(b_2o_n+j_2) \qquad (2)$$

Wherein this s denotes decoding functionality, the weight matrix for the decoder is represented by b_2 and this bias vector is denoted by variable j_2.

Algorithm-1

```
Stacked auto encoder
$o_n = c(b_1 x_n + j_1)$
$b_1$ − weight matrix, $j_1$ − bias vector
$\dot{X}_n = s(b_2 o_n + j_2)$
$b_2$ − weight matrix, $j_2$ − bias vector
$\phi(\odot)\ \arg_{\theta\theta^1} \min \frac{1}{2} \sum_{i=1}^{n} L(p^i, \dot{X}^i)$
L- Loss function
```

Algorithm-2

```
XGBoost algorithm
Initialization:
1.  Given training data from the instance space
    $R = \{(p_1, q_1), \dots, (p_k, q_k)\}$where $p_i \epsilon P$ and $q_i \epsilon q = \{-1, +1\}$.
2. Initialize the distribution $A_1(i) = \frac{1}{k}$,
Algorithm:
For ts=1,…, TS: do
    Train a learner $h_{ts} : X \rightarrow R$ using distribution $A_{ts}$ .
    Determine weight $\alpha_{ts}$ of $h_{ts}$ .
    Update the distribution over the training set:
    $A_{ts+1}(i) = \frac{A_{ts}(i) e^{-\alpha_{ts} q_{ts} h_{ts}(p_i)}}{E_{ts}}$
    Where, $E_{ts}$ is a normalization factor chosen so that $A_{ts+1}$ corres
end for
Final score:
$c(x) = \sum_{ts=0}^{TS} \alpha_{ts} h_{ts}(p)$ and H(p)=sign (c(p))
```

The parameter sets of this auto-encoder were optimized for reducing construction error (Equation 3).

$$\phi(\odot)\ \arg_{\theta\theta^1} min \frac{1}{2} \sum_{i=1}^{n} L(p^i, \dot{X}^i) \tag{3}$$

Where in this L denotes loss-function (Equation 4)

$$L\ ||p - \hat{x}||\ ^2 \tag{4}$$

As event detection model relies on this deep stacked auto-encoder to hidden layers through an unsupervised layer-wise learning algorithm. These layers were fine-tuned through a supervised method. Hence, this event detection feature fusion phase could be divided into three steps:

- To train the first auto-encoder by input data and learned feature vectors are obtained.

- The former layer feature vector utilized to be input to the next layer, and this process will be repeated until the entire training process is completed.

- After the hidden-layers were all trained, the XGBoost algorithm with cross-entropy loss function was utilized to minimize the classification loss, and weights are updated with labeled training-set, in order to achieve better fine-tuned performance results.

2.1.5. XGBoost Algorithm

Gradient Boosting utilized differentiable function-loss from weak learner for the generalization of data features. The feature loss from weak learners could be avoided by the gradient boost algorithm. At every boosting stage, learners utilized in reducing loss-function provided the current model. These boosting algorithms could be utilized in a classification task. XGBoost is a carefully parallelized optimal version of the gradient boosting algorithm. This algorithm parallelized the entire whole boosting phase and enhance the training time of event detection highly.

The training data is provided from instance space. The instance space represented by:

$R = \{(p_1, q_1), \dots, (p_k, q_k)\}$where $p_i \epsilon P$ and $q_i \epsilon q = \{-1, +1\}$.

The distribution of features of image frames from feature extraction was initialized by (Equation 5):

$$A_1(i) = \frac{1}{k} \tag{5}$$

In this XGBoost algorithm, for preventing data loss from weak learners, the weak learner is trained well through distribution A_{ts}.

The weight function (α_{ts}) is determined for each feature of h_{ts}.

The distribution is updated across training data-set (Equation 6)

$$A_{ts+1}(i) = \frac{A_{ts}(i) e^{-\alpha_{ts} q_{ts} h_{ts}(p_i)}}{E_{ts}} \tag{6}$$

Wherein defines the normalization factor, selected hence this A_{ts+1} would be distribution.

Then the final features score obtained represented by (Equation 7):

$$\mathrm{c(x)} = \sum_{\mathrm{ts=0}}^{\mathrm{TS}} \alpha_{\mathrm{ts}}\, \mathrm{h_{ts}(p)\ and\ H(p)=sign\ (c(p))} \tag{7}$$

2.1.6. Extreme Gradient Boosting Classifier–Feature Classification Handling Imbalanced Dataset and Loss

(XGBoost) Extreme boost Gradient Boosting represents a tree-based algorithm, which increases in popularity peculiar in the classification of data in recent decades. This algorithm proved a highly effective technique in data classification, such that it is more end-to-end scalable boosting model utilized in machine learning for regression and classification processes. With the substitution of FC layers as Softmax or Sigmoid functions, these layers are utilized for the extraction process. After this, the features corresponding to the image will pass via both the feature extraction models (VGG-16 and ResNet-50). Layers of individual models undertake their learning operations from features. Finally, all features return to the FC layer. These features are attained from the network and later fused by Stacked Auto Encoder. This is then passed into the proposed ML classifier. The scalable nature of this XGBoost classifier possesses the greater potential to get applied to the classification process, generally in case of handling label imbalanced and large-scale data.

At first, the tree ensemble technique for feature classification trees having a set of u_E^i | with I Є 1…. U nodes. The finalized prediction outcomes of class-label $\hat{y}_i$ were computed on the basis of the total prediction of events at leaf node c_u for every tree u^{th}. As defined in Equation 8 below:

$$\hat{y}_i = \phi(p_i) = \textstyle\sum_{u=1}^{u} c_{u(p_i)},\ \ c_u \epsilon F \tag{8}$$

Where $\mathrm{p_i}$ represents the training set and this F denotes the group all U scores for the ensemble classification method. Then this regularization step was employed in improving outcomes of event prediction as represented in Equation 9 below:

$$OLF(\phi) = \sum_i olf(\hat{y}_i, y_i) + \sum_u \Omega(c_u) \tag{9}$$

Algorithm-3

Extreme Gradient Boosting Classifier

$\hat{y}_i = \phi(p_i) = \sum_{u=1}^{u} c_{u(p_i)},\ \ c_u \epsilon F$

p_i- Training set

$\mathrm{OLF}(\phi) = \sum_i \mathrm{olf}(\hat{y}_i, y_i) + \sum_u \Omega(c_u)$,

OLF- optimal loss function, Ω – model complexity

$\Omega(c) = \gamma TS + \frac{1}{2}\lambda \sum_{j=1}^{ts} b_j^2$,

ts- Leaves in the tree, b- value of weights

$\acute{L}^{(ts)} = \sum_{i=1}^{n} [s_i c_{ts}(p_i) + \frac{1}{2} u_i c_{ts}^2 (p_i)] + \Omega(c_{ts})$

$= \sum_{i=1}^{n} [s_i\, c_{ts}(p_i) + \frac{1}{2} u_i c_{ts}^2 (p_i)] + \gamma TS + \frac{1}{2}\lambda \sum_{j=1}^{TS} b_j^2$

$= \sum_{j=1}^{TS} [(\sum_{i \epsilon I_j} s_i)\, w_j + \frac{1}{2} (\sum_{i \epsilon I_j} u_i + \lambda)\, b_j^2\ ] + \gamma TS$

$I_j = \{i | q(pi) = j\}$ - instance of leaf t, s_i and n_i order gradient statistics of the loss function

$s_i = \frac{dl(\hat{y}_i^{(ts-1)}, p_{ts})}{d\hat{y}_i^{(ts-1)}}$

$n_i = \frac{d^2 l(\hat{y}_i^{(ts-1)}, p_{ts})}{d\left(\hat{y}_i^{(ts-1)}\right)^2}$

Optimal loss function b_j^* of leaf j weight-loss function

$b_j^* = \frac{\sum_{i \epsilon I_j} s_i}{\sum_{i \epsilon I_j} n_i + \lambda}$

q- Quality of a tree structure

$O\acute{L}F^{(ts)}(q) = -\frac{1}{2} \sum_{j=1} TS \frac{(\sum_{i \epsilon I_j} s_i)^2}{\sum_{i \epsilon I_j} n_i + \lambda} + \gamma TS$

Set of left I_L and right I_R node after the splitting

$OLF_{split=\frac{1}{2}} [\frac{((\sum_{i \epsilon I_{OLF}} s_i)^2}{\sum_{i \epsilon I_j} n_i + \lambda} + \frac{(\sum_{i \epsilon I_R} s_i)^2}{\sum_{i \epsilon I_R} n_i + \lambda} + \frac{(\sum_{i \epsilon I} s_i)^2}{\sum_{i \epsilon I} n_i + \lambda}] - \gamma$

Where ***I*** = $I_R \cup I_L$

In this equation, olf denotes the weight loss functionality, defined by computing error difference among predicted class labels and target $\mathrm{y_i}$ class labels. The second section, performed penalization Ω on the complexity of the model in preventing the overfitting issue in the dataset. The functionality of penalty Ω was computed through Equation 10 below:

$$\Omega(c) = \gamma TS + \frac{1}{2}\lambda \sum_{j=1}^{ts} b_j^2 \tag{10}$$

In this above equation, the configurable parameters are denoted by γ and λ such that it controls regularization degree. The variable TS points out leaves within the tree and variable b stores out feature weight values for every leaf in the tree.

After this process, Gradient Boosting (GB) is employed to rectify the classification issue of event classes with loss functionality and extended through Taylor expansion. The constant term would be eliminated to gain a simple objective at phase or step ts, such that it is computed in Equation 11 below:

$$\acute{L}^{(ts)}=\sum_{i=1}^{n}[s_i c_{ts}(p_i)+\ \tfrac{1}{2}u_i c_{ts}^2\,(p_i)]+\ \Omega(c_{ts}) \qquad (11)$$

$$=\sum_{i=1}^{n}[s_i\ c_{ts}(p_i)+\ \frac{1}{2}u_i c_{ts}^2\,(p_i)]+\ \gamma TS + \frac{1}{2}\lambda\sum_{j=1}^{TS}b_j^2 \qquad (12)$$

$$=\sum_{j=1}^{TS}[(\textstyle\sum_{i\epsilon I_j}s_i)\,w_j+\tfrac{1}{2}\ \ (\textstyle\sum_{i\epsilon I_j}u_i+\lambda)\,b_j^2\ ]+\gamma TS \qquad (13)$$

Wherein $I_j = \{i|\ q\ (p_i) = j\}$ represents the instance of leaf t, and the equation for s_i and u_i order gradient statistics of loss function were defined in Equations 14, 15 below:

$$s_i = \frac{dl(\hat{y}_i^{(ts-1)}, p_{ts})}{d\hat{y}_i^{(ts-1)}} \qquad (14)$$

$$n_i = \frac{d^2 l(\hat{y}_i^{(ts-1)}, p_{ts})}{d\left(\hat{y}_i^{(ts-1)}\right)^2} \qquad (15)$$

The optimal weight b_j^* for tree leaf j could be computed by Equation 16 below:

$$b_j^* = \frac{\sum_{i\epsilon I_j}s_i}{\sum_{i\epsilon I_j}n_i+\lambda} \qquad (16)$$

A unique function was utilized to score function in measuring the tree structure (q) quality for provided tree structure q (x_i) could be calculated through Equation 17.

$$O\acute{L}F^{(ts)}(q) = -\frac{1}{2}\sum_{j=1}TS\frac{\left(\sum_{i\epsilon I_j}s_i\right)^2}{\sum_{i\epsilon I_j}n_i+\lambda}+\gamma TS \qquad (17)$$

Typically in order to measure split nodes through employing scoring in instance group of right I_R and left I_L nodes after the splitting process are performed the reduction in data weight loss were computed in below Equation 18:

$$OLF_{split=\frac{1}{2}}\left[\frac{\left(\left(\sum_{i\epsilon I_{OLF}}s_i\right)^2\right.}{\sum_{i\epsilon I_j}n_i+\lambda}+\frac{\left(\sum_{i\epsilon I_R}s_i\right)^2}{\sum_{i\epsilon I_R}n_i+\lambda}+\frac{\left(\sum_{i\epsilon I}s_i\right)^2}{\sum_{i\epsilon I}n_i+\lambda}\right]-\gamma \qquad (18)$$

Where in I points to I_R U I_L

2.1.7. Modified XGBoost Weight-Loss Function

The modified XGBoost algorithm was adaptable of handling larger dataset, the algorithm achieved yields out the target variable estimate by establishing decision tree series and allocating every leaf node with quantized weight functionality. This loss function referred to as the optimal loss function, aimed to deal with classification issues as the data features are trained deeply layer by layer, there may be a change to the loss of the data from the original image frame. The intensity of outcomes may be dissimilar: like it may interpret '0' instead of '1' different values in an array matrix representing image frames.

The XGBoost algorithm achieves an estimate of the target variable by establishing a series of decision trees and assigning each leaf node a quantized weight.

The initialization of the distribution is performed by (Equation 19):

$$A_{ts+1}(i) = \frac{A_{ts}(i)e^{-\alpha_{ts}q_{ts}h_{ts}(p_i)}}{E_{ts}} \qquad (19)$$

Such that iteration count is ts ranging from 1 …TS

In this XGBoost algorithm, this regularization term is affixed to the loss function that takes the complexity and accuracy of the event detection model at the same time. The group of prediction functions in this model was learned and trained through reducing below total loss function (Equation 20).

$$OLF(\Phi) = \sum_i olf(\hat{y}_i, y_i) + \sum_u \Omega(c_u) \qquad (20)$$

In this above equation, OLF represents the loss functionality by using cross entropy-loss function, pointing to model fitness seems as a measurement of variations between predictive and real values. The complexity of the model is denoted by Ω. The loss functionality utilized is square-loss such that olf $(\hat{y}_i, y_i) = (\hat{y}_i, y_i)^2$. Using this $\Omega = \gamma TS + \frac{1}{2}\lambda\sum_{j=1}^{TS}b_j^2$ in measuring the model complexity wherein λ and γ were tuning parameters.

Algorithm-II: OWu-XGBoost-SVM

```
Step-1: Initialization
Training data from instance space (TS)
$S = \{(p_1, q_1), \dots, (p_n, q_n)\}$ where $p_i \epsilon P$ and $q_i \epsilon Q = \{-1, +1\}$
Step-2: Initialize distribution $Dist_1(i) = \frac{1}{n}$
For ts=1….TS: do
Train a weak-learner $u_{ts}: A \rightarrow R$
Step 3: Find the weight $\delta_{ts}$ of $u_{ts}$
Step-4: Update distribution upon the training set
$Dist_{ts+1}(i) = \frac{Distri_{ts}(i)e^{-\delta_{ts} b_i u_{ts}(p_i)}}{E_{ts}}$
Wherein, $N_{ts}$-chosen normalization factor, $Dist_{ts+1}$-distribution
End for
Step-5: Overall score
$c(p) = \sum_{ts=0}^{TS} \delta_{ts} w_{ts}(p)$ and $N(p) = sign(c(p))$
//SVM classifier:
Input: Extracted Significant Features with medical data of the patie
Step-6: Begin
For each train data of a classification ‘ ’
Create an ′N ′ number of base SVM classifier
$N = (P1, P2, P3, \dots PN)$
Step-7: Initialize weight ‘W ’for each base classifier
$\vartheta - -$signifies the weight vector, $-$
                    $\rightarrow P_i$ represents the data in classification , and b
                    $- -$ is a bias.
$\vartheta. P_1 + b > 0$
$\vartheta. P_1 + b < 0$
Step 8: Determine the negative gradient
Fit a base classifier to a negative gradient using
$d = \varphi_1, \varphi_2(\frac{2}{||\omega||})$
$\gamma i = sign(\omega * Pi + b) = sign(\sum_{i=1}^{n} \vartheta * \mu_i Pi + b)$
Step-9: Update weights ‘ W’ of base classifiers using
$w_i = \sum_{i=1}^{n} \emptyset_i(P)$
Step 10: Discover the best gradient descent step − size
$\xi = (ACi - \emptyset(P))^2$
Step-11: Update the model as a strong classifier
$\emptyset_i(P) = (P, (ACi - \rho(Pi))$
Strong classifier provides classification results‘ ’
Step-12: If results is $p == 0$ then
The patient is classified as normal, beneign
else results are $p = ! 0$
The patient is classified as Invasive, Insitu
 Endif
 End for
End
```

3. Results

3.1. Comparative Assessment of VGG-16 and ResNet-50 Feature Extraction Learning with Other Techniques

The Figure 5 illustrates the performance assessment of the proposed model, in terms of accuracy metric, it was assessed by comparing the event detection outcomes of different methods, including CNN-based attention model, Distribution Attention Supervision model, CNN-LSTM method, CNN aggregated method, existing techniques in handling the different event features, and event concepts. The results of the proposed event detection mechanism are higher in graphical representation.

Table 3 enumerated that the proposed model exhibited to detect the event detection, like it belongs to which class was determined by the proposed model with 97.58 % of accurate data. The other methods determine the events, with an accuracy rate of 85.50 % (CNN-based attention model), Distribution Attention Supervision model, 91.10 % (CNN-LSTM method), CNN-aggregated method, and existing methods, 96.45% in event detection. The higher accuracy in event detection defines the efficiency of the framework.

Similar to this, another performance evaluation of the proposed framework is in determining how far the model is accurate in finding the appropriate event detection from feature extraction using VGC and ResNet model and training of features using optimal loss XGBoost classifier, with Deep stacked auto-encoder algorithm with a comparison of other

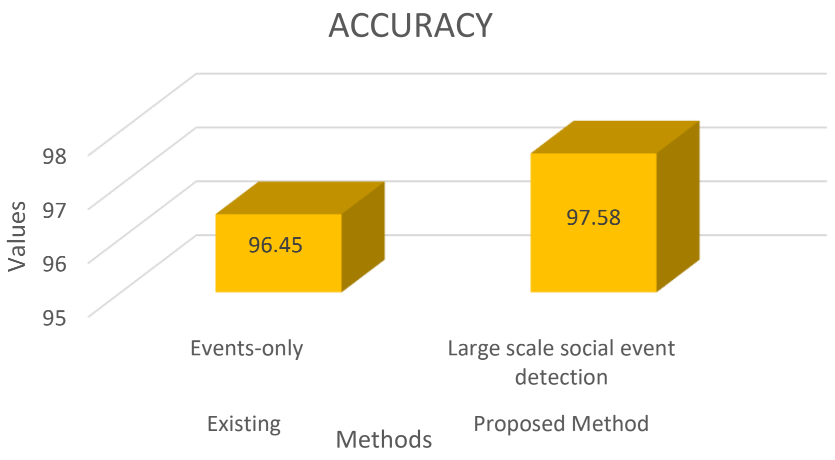


Figure 5. Comparative analysis of accuracy rate in event detection in PEC dataset [35]

Table 3. Accuracy metrics values of proposed event detection classifier with other existing methods

Method	Features	PEC ACCURACY
CNN-based Attention Model [36]	Events - only	85.5
CNN-based Attention Model	Events + importance	87.9
Distribution Attention Supervision [37]	Events - only	
CNN-LSTM Model [38]	Events + importance (extended)	91.1
CNN-aggregated	Events - only	
Existing	Events - only	**96.45**
Proposed Method	Large scale social event detection	**97.58**

algorithms or methods. From graphical representation, Figure 6 above, it is clearly explicated that the proposed method exhibited a higher accuracy level of event detection.

Table 4 describes the different accuracy rates of different event detection methods implementing different algorithms such as mmLDA with SVM feature classification, mm-SLDA classifier, BMM-SLDA model, and SLDA model classifier. The proposed XGBoost classifier with loss function and VGG and ResNet feature extraction for event detection explicated event detection outcomes with 97.50 % accuracy rate. The resultant outcomes depict the outperforming performance of the proposed model compared with other event detection techniques.

3.2. Comparative Performance Analysis of Proposed Feature Classifier in Event Detection

Another assessment is to bring out the performance results of the proposed model using VGG-16 and ResNet-50 features extraction with Optimal Loss XGBoost classifier by comparing with other classifiers in event classification.

Figure 7 propounded the performance analysis of the proposed event classification method with existing techniques in terms of accuracy metric. The photo event collection dataset was utilized to test the performance of the event detection mechanism. From the figure, it is clear that the proposed event classification and prediction method using the Optimal loss function XGBoost classifier classifies and determines events features relying on the Photo

Table 4. Accuracy measurement of the proposed method in event detection with conventional approaches [39]

Methods	Accuracy
mmLDA+SVM	0.724
SLDA(visual)	0.312
mm-SLDA	0.722
BMM-SLDA	0.835
Proposed-RF with LF	**0.975**

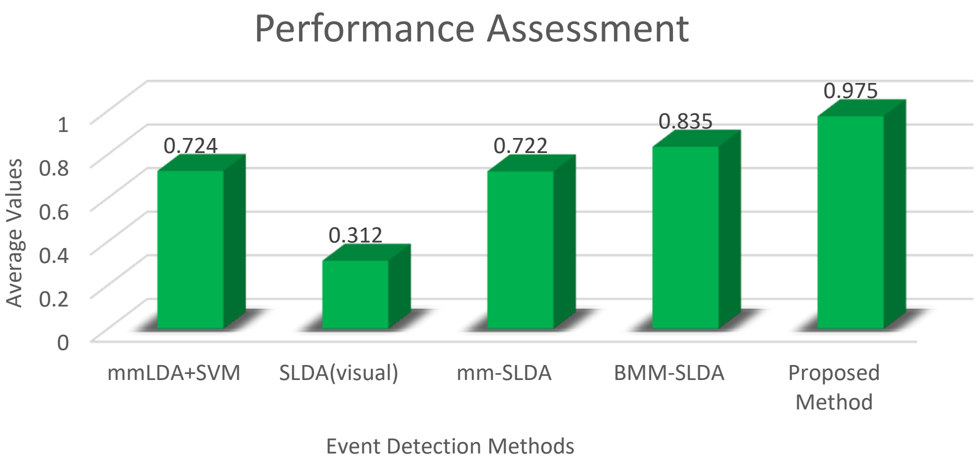


Figure 6. Performance analysis of the proposed model in terms of accuracy

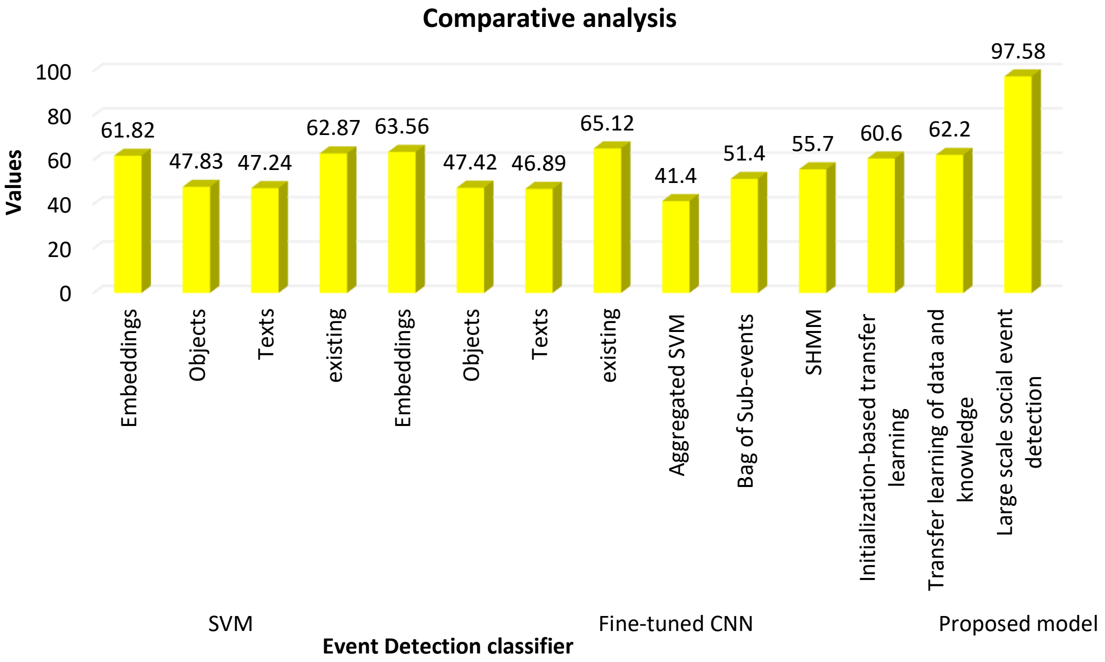


Figure 7. Comparative assessment of the proposed method with existing event detection technique (accuracy metric) [40]

event collection dataset more efficiently than other existing classifiers categorizing embedding, objects, texts, and so on features for event detection.

Table 5 enumerates the classification accuracy rate of deep models using the SVM classifier, Fine-tuned CNN, Aggregated SVM, Bag of sub-events, Initialized Base (TL) Transfer learning, and SHMM with the proposed event classification method. The method extracts and classifies Embedding features, Object features, Text features using SVM classifier with 61.82%, 47.83%, 47.24, and 62.87 rates of accuracy. The same features using Fine-tuned CNN model classify the event features with 63.56%, 47.52%, 46.89, and 65.12% rates of accuracy. The aggregated SVM classifier, Bag-of events classifier, SHMM classifier, TL based classifier poses classify features with accuracy rate of 41.4%, 51.40%, 55.70%, and 62.20%. All the features classified by those existing deep models classifier bring out outcomes lesser than 70%. But deep model of VGG-16 and ResNet-50 features extraction with the Optimal Loss XGBoost classifier using Deep stacked auto-encoder feature fusion in large-scale social event detection dataset showed 97.58 rate of accuracy in classifying the features based on event classes. The higher accuracy reveals the outstanding performance of the proposed method.

Table 5. Comparative analysis of the proposed Deep learning based optimal loss XGBoost classifier with other classifiers in terms of accuracy

Classifier	Features	Deep models
SVM	Embedding	61.82
	Objects	47.83
	Texts	47.24
	existing	62.87
Fine-tuned CNN	Embedding	63.56
	Objects	47.42
	Texts	46.89
	existing	65.12
Aggregated SVM [5]		41.4
Bag of Sub-events [5]		51.4
SHMM [5]		55.7
Initialization-based transfer learning [31]		60.6
Transfer learning of data and knowledge [31]		62.2
Proposed model	**Large scale social event detection**	**97.58**

4. Discussions

The entire comparative assessment of the proposed method, with other existing event detection mechanisms, including the SVM classifier, Aggregated SVM method, Transfer learning, Fine-tuned CNN model elucidated the efficacy of the optimal loss XGBoost classifier with VGG-16 and ResNet-50 feature extraction learning. Similarly, in measuring the performance score of the proposed model in accordance with the accuracy metric in Figures 5, 6, 7 delineated that the proposed Deep model event detection classifies the features with 97.50 % rate of accuracy in event dataset than other methods such as mmLDA with SVM classifier, Mm-SLDA, BMM-SLDA and SLDA classifier. The higher accuracy rate in inhibiting all the features in deep layer learning in the proposed model reveals the outstanding performance of the proposed event detection method. Although the proposed system has shown better outcomes, it consumes time to perform feature extraction and training. Besides, predicting a single image requires further processes.

5. Conclusion

In this study, the implementation of VGG-16 and ResNet-50 neural network learns the feature extraction in more deep layers with smaller convolutional filters that yield distinctive features and collaboration decreases vanishing gradient data loss issue and maximizes the learning rate. Then features are integrated through feature fusion, aided by deep stacked auto-encoders, with encoder and decoder input outputs feeds to handle data imbalance issues, oversee the layer-by-layer data, back-propagating input data and output in each layer. The proposed study gains higher accuracy in data classification through employing the optimal loss function with the XGBoost model classifier, such that categorical loss function and weight variants inhibited each feature and classified. Hence, the misclassification issue dealt with the usage of the optimal loss function. The performance assessment of the proposed model was enumerated by assessing event detection accuracy of 97.58% in comparison with other conventional methods, differentiated to various feature labels in different datasets. The results explicated outperforming metrics with the accuracy rate of more than other existing event detection models.

References

1- Weikang Wang *et al.*, "Frequency disturbance event detection based on synchrophasors and deep learning." *IEEE Transactions on Smart Grid,* Vol. 11 (No. 4), pp. 3593-605, (2020).

2- Peng Wu, Jing Liu, and Fang Shen, "A deep one-class neural network for anomalous event detection in complex scenes." *IEEE transactions on neural networks and learning systems,* Vol. 31 (No. 7), pp. 2609-22, (2019).

3- Anitha Ramchandran and Arun Kumar Sangaiah, "Unsupervised deep learning system for local anomaly event detection in crowded scenes." *Multimedia Tools and Applications,* Vol. 79 (No. 47), pp. 35275-95, (2020).

4- Shubhangi Kale and Raghunathan Shriram, "Suspicious Activity Detection Using Transfer Learning Based ResNet Tracking from Surveillance Videos." in *International Conference on Soft Computing and Pattern Recognition,* (2020): *Springer*, pp. 208-20.

5- Zied Mnasri, Stefano Rovetta, Francesco Masulli, and Alberto Cabri, "Dealing with Uncertainty in Anomalous Audio Event Detection Using Fuzzy Modeling." in *UK Workshop on Computational Intelligence*, (2021): *Springer*, pp. 496-507.

6- Sarah Almeida Carneiro, Silvio Jamil Ferzoli Guimarães, and Hélio Pedrini, "High-Level Descriptors for Fall Event Detection Supported by a Multi-Stream Network." *International journal of electrical and computer engineering systems*, Vol. 12 (No. 1), pp. 11-21, (2021).

7- Wenqing Chu, Hongyang Xue, Chengwei Yao, and Deng Cai, "Sparse coding guided spatiotemporal feature learning for abnormal event detection in large videos." *IEEE Transactions on Multimedia,* Vol. 21 (No. 1), pp. 246-55, (2018).

8- Long Wen, Xinyu Li, and Liang Gao, "A transfer convolutional neural network for fault diagnosis based on ResNet-50." *Neural Computing and Applications*, Vol. 32 (No. 10), pp. 6111-24, (2020).

9- Yousef I Mohamad, Samah S Baraheem, and Tam V Nguyen, "Olympic Games Event Recognition via Transfer Learning with Photobombing Guided Data Augmentation." *Journal of Imaging,* Vol. 7 (No. 2), p. 12, (2021).

10- Manoj Kumar Panda, Akhilesh Sharma, Vatsalya Bajpai, Badri Narayan Subudhi, Veerakumar Thangaraj, and Vinit Jakhetiya, "Encoder and decoder network with ResNet-50 and global average feature pooling for local change detection." *Computer Vision and Image Understanding,* Vol. 222p. 103501, (2022).

11- Samir Bouindour, Mohamad Mazen Hittawe, Sandy Mahfouz, and Hichem Snoussi, "Abnormal event detection using convolutional neural networks and 1-class SVM classifier." in *8th International Conference on Imaging for Crime Detection and Prevention (ICDP 2017),* (2017): *IET*, pp. 1-6.

12- Guifang Liu, Huaiqian Bao, and Baokun Han, "A stacked autoencoder-based deep neural network for achieving gearbox fault diagnosis." *Mathematical Problems in Engineering,* Vol. 2018(2018).

13- Xiaojun Chang, Zhigang Ma, Yi Yang, Zhiqiang Zeng, and Alexander G Hauptmann, "Bi-level semantic representation analysis for multimedia event detection." *IEEE transactions on cybernetics,* Vol. 47 (No. 5), pp. 1180-97, (2016).

14- Zhandong Wang, Shuqin Lou, Sheng Liang, and Xinzhi Sheng, "Multi-class disturbance events recognition based on EMD and XGBoost in φ-OTDR." *IEEE Access,* Vol. 8pp. 63551-58, (2020).

15- Fabio Vesperini, Leonardo Gabrielli, Emanuele Principi, and Stefano Squartini, "Polyphonic sound event detection by using capsule neural networks." *IEEE Journal of Selected Topics in Signal Processing,* Vol. 13 (No. 2), pp. 310-22, (2019).

16- Joshua Emoto and Yutaka Hirata, "Lightweight convolutional neural network for image processing method for gaze estimation and eye movement event detection." *IPSJ Transactions on Bioinformatics,* Vol. 13pp. 7-15, (2020).

17- Mrutyunjaya Sahani and Pradipta Kishore Dash, "Deep convolutional stack autoencoder of process adaptive VMD data with robust multikernel RVFLN for power quality events recognition." *IEEE Transactions on Instrumentation and Measurement,* Vol. 70pp. 1-12, (2021).

18- Saman Sarraf and Mehdi Noori, "Multimodal deep learning approach for event detection in sports using Amazon SageMaker." *AWS Machine Learning Blog,* (2021).

19- Ali Karimi, Ramin Toosi, and Mohammad Ali Akhaee, "Soccer Event Detection Using Deep Learning." *arXiv preprint arXiv:2102.04331,* (2021).

20- Hansi Hettiarachchi, Mariam Adedoyin-Olowe, Jagdev Bhogal, and Mohamed Medhat Gaber, "Embed2Detect: Temporally clustered embedded words for event detection in social media." *Machine Learning,* Vol. 111 (No. 1), pp. 49-87, (2022).

21- Yijian Zhou, Han Yue, Qingkai Kong, and Shiyong Zhou, "Hybrid event detection and phase-picking algorithm using convolutional and recurrent neural networks." *Seismological Research Letters,* Vol. 90 (No. 3), pp. 1079-87, (2019).

22- Mohamad Mazen Hittawe, Shehzad Afzal, Tahira Jamil, Hichem Snoussi, Ibrahim Hoteit, and Omar Knio, "Abnormal events detection using deep neural networks: application to extreme sea surface temperature detection in the Red Sea." *Journal of Electronic Imaging,* Vol. 28 (No. 2), p. 021012, (2019).

23- Tong Li, Xinyue Chen, Fushun Zhu, Zhengyu Zhang, and Hua Yan, "Two-stream deep spatial-temporal auto-encoder for surveillance video abnormal event detection." *Neurocomputing,* Vol. 439pp. 256-70, (2021).

24- Yangyang Li, Jun Li, Hao Jin, and Liang Peng, "Focusing Attention across Multiple Images for Multimodal Event Detection." in *ACM Multimedia Asia*, (2021), pp. 1-6.

25- Yaxiang Fan, Gongjian Wen, Deren Li, Shaohua Qiu, and Martin D Levine, "Early event detection based on dynamic images of surveillance videos." *Journal of Visual Communication and Image Representation,* Vol. 51pp. 70-75, (2018).

26- Mariana Iuliana Georgescu, Radu Tudor Ionescu, Fahad Shahbaz Khan, Marius Popescu, and Mubarak Shah, "A background-agnostic framework with adversarial training for abnormal event detection in video." *IEEE Transactions on Pattern Analysis and Machine Intelligence,* Vol. 44 (No. 9), pp. 4505-23, (2021).

27- Anastasia Sokolova, Yuri Uljanitski, Airat R Kayumov, and Mikhail I Bogachev, "Improved online event detection and differentiation by a simple gradient-based nonlinear transformation: Implications for the biomedical signal and image analysis." *Biomedical Signal Processing and Control,* Vol. 66p. 102470, (2021).

28- Kaavya Kanagaraj and GG Priya, "A new 3D convolutional neural network (3D-CNN) framework for multimedia event detection." *Signal, Image and Video Processing,* Vol. 15 (No. 4), pp. 779-87, (2021).

29- Michael Hertkorn and ZF Friedrichshafen AG, "Few-shot bioacoustic event detection: Don't waste information." ed: *June*, (2022).

30- Zhihui Li, Lina Yao, Xiaojun Chang, Kun Zhan, Jiande Sun, and Huaxiang Zhang, "Zero-shot event detection via event-adaptive concept relevance mining." *Pattern Recognition,* Vol. 88pp. 595-603, (2019).

31- Hao Song, Che Sun, Xinxiao Wu, Mei Chen, and Yunde Jia, "Learning normal patterns via adversarial attention-based autoencoder for abnormal event detection in videos." *IEEE Transactions on Multimedia,* Vol. 22 (No. 8), pp. 2138-48, (2019).

32- Yunhao Liang, Yanhua Long, Yijie Li, Jiaen Liang, and Yuping Wang, "Joint framework with deep feature distillation and adaptive focal loss for weakly supervised audio tagging and acoustic event detection." *Digital Signal Processing,* Vol. 123p. 103446, (2022).

33- Changjun Fan and Fei Gao, "A new approach for smoking event detection using a variational autoencoder and neural decision forest." *IEEE Access,* Vol. 8pp. 120835-49, (2020).

34- Marco Willi *et al.*, "Identifying animal species in camera trap images using deep learning and citizen science." *Methods in Ecology and Evolution,* Vol. 10 (No. 1), pp. 80-91, (2019).

35- Tamar Glaser, Emanuel Ben-Baruch, Gilad Sharir, Nadav Zamir, Asaf Noy, and Lihi Zelnik-Manor, "PETA: Photo Albums Event Recognition using Transformers Attention." *arXiv preprint arXiv:2109.12499,* (2021).

36- Cong Guo, Xinmei Tian, and Tao Mei, "Multigranular event recognition of personal photo albums." *IEEE Transactions on Multimedia*, Vol. 20 (No. 7), pp. 1837-47, (2017).

37- Seungtaek Choi, Haeju Park, and Seung-won Hwang, "Meta-supervision for attention using counterfactual estimation." *Data Science and Engineering,* Vol. 5 (No. 2), pp. 193-204, (2020).

38- Yufei Wang, Zhe Lin, Xiaohui Shen, Radomír Mech, Gavin Miller, and Garrison W Cottrell, "Recognizing and curating photo albums via event-specific image importance." *arXiv preprint arXiv:1707.05911,* (2017).

39- Shengsheng Qian, Tianzhu Zhang, Changsheng Xu, and M Shamim Hossain, "Social event classification via boosted multimodal supervised latent dirichlet allocation." *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM),* Vol. 11 (No. 2), pp. 1-22, (2015).

40- Andrey V Savchenko, "Event recognition with automatic album detection based on sequential processing, neural attention and image captioning." *arXiv preprint arXiv:1911.11010,* (2019).