

Intelligent Ammunition Detection and Classification System Using Convolutional Neural Network

Gulzar Ahmad¹, Saad Alanazi², Madallah Alruwaili², Fahad Ahmad^{3,6}, Muhammad Adnan Khan^{4,*},
Sagheer Abbas¹ and Nadia Tabassum⁵

¹School of Computer Science, NCBA&E, Lahore, 54000, Pakistan

²College of Computer and Information Sciences, Jouf University, Sakaka, Aljouf, 72341, Saudi Arabia

³Department of Basic Sciences, Jouf University, Sakaka, Aljouf, 72341, Saudi Arabia

⁴Riphah School of Computing & Innovation, Faculty of Computing, Riphah International University,
Lahore Campus, Lahore, 54000, Pakistan

⁵Department of Computer Science, Virtual University of Pakistan, Lahore, 54000, Pakistan

⁶Department of Computer Sciences, Kinnaird College for Women, Lahore, 54000, Pakistan

*Corresponding Author: Muhammad Adnan Khan. Email: adnan.khan@riphah.edu.pk

Received: 05 November 2020; Accepted: 27 December 2020

Abstract: Security is a significant issue for everyone due to new and creative ways to commit cybercrime. The Closed-Circuit Television (CCTV) systems are being installed in offices, houses, shopping malls, and on streets to protect lives. Operators monitor CCTV; however, it is difficult for a single person to monitor the actions of multiple people at one time. Consequently, there is a dire need for an automated monitoring system that detects a person with ammunition or any other harmful material. Based on our research and findings of this study, we have designed a new Intelligent Ammunition Detection and Classification (IADC) system using Convolutional Neural Network (CNN). The proposed system is designed to identify persons carrying weapons and ammunition using CCTV cameras. When weapons are identified, the cameras sound an alarm. In the proposed IADC system, CNN was used to detect firearms and ammunition. The CNN model which is a Deep Learning technique consists of neural networks, most commonly applied to analyzing visual imagery has gained popularity for unstructured (images, videos) data classification. Additionally, this system generates an early warning through detection of ammunition before conditions become critical. Hence the faster and earlier the prediction, the lower the response time, losses and potential victims. The proposed IADC system provides better results than earlier published models like VGGNet, OverFeat-1, OverFeat-2, and OverFeat-3.

Keywords: CCTV; CNN; IADC; deep learning; intelligent ammunition detection; DnCNN



This work is licensed under a Creative Commons Attribution 4.0 International License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

1 Introduction

In the current era, security has become a vital issue. Under current prevailing conditions of poor security system, which has caused fear among people, everyone wants to secure his resources; premises; his organization's employees and clients. However, CCTV systems are becoming more popular for providing additional security. They are typically installed in many public places such as roads, highways, offices, housing complexes, and shopping malls. Generally, CCTV systems can see weapons or other harmful items in a person's hand. However, the potential threat must be identified by a remote operator, who then may trigger a police response [1]. Moreover, the images recorded by CCTV might not be examined until after a criminal incident. For this reason, the system proposed in this study is designed to enable CCTV cameras to detect weapons and potential threats in real-time.

The primary aim of this research was to minimize or eliminate the threat to security posed by weapons (e.g., pistols, automatic weapons, and knives) or explosives frequently used in criminal activities. In this research, we proposed an automated system to detect any dangerous weapons in the hand of an attacker or a terrorist. This system sounds an alarm that alerts the CCTV operator, who immediately informs the police or other agencies. It can help citizens before conditions worsen or the crime is carried out.

The rate of crimes involving weapons is of increasing global concern, specifically in those countries where gun ownership is legal. Initial detection is needed to detect weapons earlier and allow law enforcement agencies to take immediate action. One of the advanced solutions to this problem is to supplement surveillance or control CCTV cameras using an automatic system to detect a pistol, gun, or revolver. It sounds an alarm once it detects any harmful objects in an attacker's hand. CNN is also used to detect a firearm using a video recorder [2].

The ammunition detection system includes a firearm and a device for communication. The firearm contains a transceiver circuit, which detects the release of an ammunition round. The transceiver circuit produces an electromagnetic signal synchronized with the ammunition discharge. The communication device, which may be mobile or any handheld device like a radio set, detects the electromagnetic signal. The modes of communication can be single, half, or full way. The communication device is paired with a geographic location sensor, for example, a GPS receiver [3]. The communication device contains software that generates an informational message with geographic information transmitted to a remote device. Later, a converted image is obtained. Another visible light image, obtained using a transforming system, is included in this transformed image [4]. The object of interest has, therefore, been identified in the field [5]. On identification, an indicator shows the object's location in the area of interest.

The techniques for monitoring a firearm and generating a warning when the firearm is outside the designated permit area defined in various scenarios. A location device, such as a Global Positioning System (GPS) receiver coupled to a transmitter, is connected to or associated with a firearm. The identified location of the weapon is transmitted to a location service module via the location device. The position service module compares at least one designated authorized location with the current firearm location and confirms that the location of the firearm is not within the authorized permit area. In response, an alert is generated. It is a multi-modal protection technique that will detect any concealed metallic (weapon or shrapnel), non-metallic (explosives or Improvised Explosive Device (IED)), or radioactive nuclear threats. Furthermore, long-range facial recognition of potential terrorists can be carried out by the security checkpoint to detect them. The security checkpoint integrates many technologies for detecting threats into a single checkpoint

designed to be stable and see a wide range of threats, including concealed weapons, explosives, bombs, and other threats [6].

The importance of security can never be neglected, especially when technological advances have broadened criminals' opportunities to commit crimes. After evaluating their security issues, many corporations have started investing in systems to secure their facilities. New visual systems that include handheld and web platforms to reduce response times are required to enhance security. It emphasizes perceiving the user's perceived load and working memory load efficiency. Visualizations are optimized when the perceived load is reduced, and working memory increases [7]. In Cloud computing, security-as-a-service is the most significant change in the field of information and corporate security. The accuracy of live streaming data depends on several factors.

Consequently, many high-technology security systems are available. The Security Information and Event Management (SIEM) system was developed to collect, analyze, aggregate, normalize, stock, and purify event logs. They also correlate data from traditional security systems such as intrusion detection/avoidance, firewalls, anti-malware, and others installed in both the host and network domains [8].

2 Previous Studies

Almotaeryi conducted research on automated CCTV surveillance [9] in which he compared different solutions to data augmentation in image classification using deep learning. Perez et al. developed a method in which a neural network could learn augmentations to improve classification process, called neural augmentation. There is an extensive discussion of this technique's pros and cons in different datasets [10]. Deperlioğlu analyzed an effective and successful method to diagnose diabetic retinopathy from retinal fundus images using image processing and deep learning techniques. CNN is used to classify images [11]. In the information communication system, the denoising of the image is a modern and practical approach due to the image-filtering algorithm. However, that algorithm is not always efficient for the nature of the noise spectrum. Sheremet et al. presented the possibilities of denoising using the convolutional neural network while transferring the graphical content in the information communication system proposed in this study. They concluded that using a denoising convolutional neural network creates the correct signal but sends noisy images [12].

Zhang et al. [13] presented the feedforward De-noising Convolutional Neural Network (DnCNN) as the primary source of progress in deep architecture, learning algorithms, and regularization techniques. Training processes accelerate because of residual learning and batch normalization strategies. Zhang et al. have presented a DnCNN model with a residual learning strategy that accommodates many images of denoising processes. Cha et al. [14] discussed a vision-based technique for recognizing concrete cracks without calculating imperfect features with the help of CNN architecture. However, CNN can learn image features automatically. This technique works for removing features without the help of Image Processing Techniques (IPT) and analyzes the efficiency of CNN using Traditional Sobel and Canny edge recognition techniques, which are significantly better.

Yang et al. [15] presented a new technique for super-resolution called multi super-resolution convolutional neural network. The development of GoogLeNet architecture inspired it. This method uses parallel convolution filters of various sizes and achieves low-resolution license plate imagery, with a concatenation layer that blends the features. Finally, this method rebuilt the high-resolution image using nonlinear mapping.

Tsoutsas et al. [16] are working on artistic styles using a neural algorithm that can separate and recombine the image material and natural image style. Recently, Leon A. Gatys and Alexander S. Eickhoff described Image Style Transfer Using Convolutional Neural Networks and feature extraction. Handa et al. [17], Simonyan et al. [18] worked on convolutional network depth and its impact on the precision of large-scale image settings. They used tiny (3×3) convolutional filters, which improved earlier art configurations by pushing depth to 16–19 weight layers.

Deep learning techniques play a vital role as an essential alternative, overcoming the difficulties of feature-based approaches. Araújo et al. [19] clarified a method that classified the hematoxylin and eosin-stained breast biopsy images using CNN. A faster R-CNN model has been trained on the bigger datasets [20]. In the last two years, deep learning methods have been improved rapidly for general object detection. Various methods of facial recognition are still based on R-CNN, resulting in limited accuracy and processing speed. Jiang et al. [21] analyzed the Faster CNN implementation, which produced impressive results in various object detection benchmarks [17,22].

Automated Intelligence Approaches have recently been used in various scientific fields such as medical domains, Smart cities, and health [23–27]. CNN has had a significant impact in various fields such as medical imaging, agriculture, smart home, smart transportation, security, and traffic law violation detection [28–32].

3 Proposed Intelligent Ammunition Detection and Classification (IADC) System

The proposed Intelligent Ammunition Detection and Classification (IADC) system uses a Convolutional Neural Network. Fig. 1 illustrates the acquired image stream from various CCTV cameras at different locations. These CCTV cameras transmit captured images to the object layer through the Cloud. Due to moving objects, the captured images may be blurred or noisy. To convert the captured images into high-quality images, a preprocessing layer is required for image enhancement. The object layer further sends the images to the preprocessing layer. The preprocessing layer sends these image streams to the Convolution Neural Network model. The CNN model classifies the object as either with ammunition or without ammunition. If ammunition is detected, the model notifies the observer of the object and sounds an alarm. If no weapon or ammunition is detected, there is no alarm.

3.1 Sensing Layer

There are N-sensing layers located in different positions, and each layer contains multiple cameras. The N-sensing layers transmit captured image streams to the object layer through the Cloud.

3.2 Object Layer

The captured image stream needs to be stored in a specific location. Image streams coming from different sensing layers are stored in object layers. The object layers combine all stream data inputs at a single point.

3.3 Pre-Processing Layer

Input image stream may contain noise and blurriness as a result of low quality. It consists of raw shape data that cannot produce good results in image classification. The preprocessing layer transforms the raw images into high-quality images by removing noise and blurriness. There are

different filters used to remove this noise and blurriness, which are the inputs of CNN. Fig. 2 shows the preprocessing process of the input raw data stream.

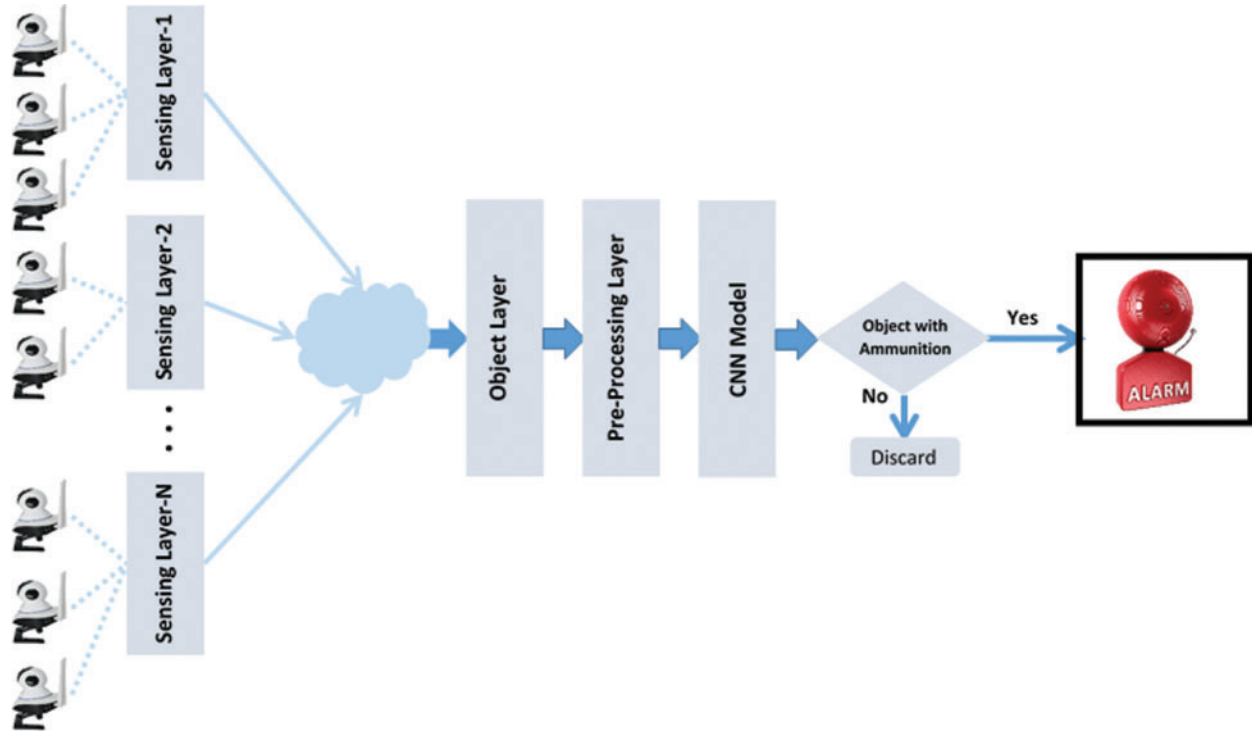


Figure 1: Intelligent ammunition detection & classification (IADC) system using a convolutional neural network



Figure 2: Pre-processing image data

3.3.1 Image Noise Model

Image streaming may be blurred or noisy. The additive and multiplicative Noisy Image Model has been clarified in Eqs. (1), (2).

$$\hat{\Psi}(x) = \Phi(x) + \Psi(x) \tag{1}$$

$$\hat{\Psi}(x) = \Phi(x) * \Psi(x) \tag{2}$$

where, $\Psi(x)$ is the original image form, $\Phi(x)$ is the noise, and $\Psi(x)$ is the noisy image.

3.3.2 Gaussian Noise

The Gaussian noise model is very popular because of its simple application. When other noise models fail, the Gaussian noise model can be applied. Eq. (3) is the mathematical representation of the Gaussian noise model.

$$\Phi(x) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \quad (3)$$

where x is the gray value, σ is the standard deviation, and μ is the mean.

3.3.3 Impulse Valued Noise

The black and white dots on the image are called salt and pepper or impulse valued noise. In Fig. 3, the centered value 200 is replaced by the value 0. Progressively, dark pixel values are replaced by white pixel values and vice versa.



Figure 3: Pixel value is corrupted by salt and pepper noise

3.4 Convolutional Neural Network Model

Today, authorities attempt to resolve most issues by seeking help from computer professionals using Artificial Intelligence (AI) methods. AI is a broad spectrum used in every aspect of life. For this purpose, machine learning, a subset of AI, is used. Machine learning is used to solve a different problem by applying various algorithms like k-nearest neighbors, linear regression, decision trees, logistic regression, Support Vector Machine, random forests, and neural networks. Deep learning is used for image classification, a subset of machine learning. In deep learning, CNN has been used. It is a powerful model for object classification. It is a network of different sequentially-connected layers.

These are convolution layers in which the convolution process occurs. It can typically have multiple convolution and pooling. Normalization layers do not necessarily follow the order. Fig. 4 shows the complete model used in the IADC system with a CNN.

To extract the feature or object for the next layer in the convolutional layer, a kernel/filter matrix is used. There are different methods to obtain the features of the images using kernel. The feature map values can be computed by the sum of the product of element-wise of input matrix and kernel. Often, a dot product is used instead of the element-wise multiplication, but this can be modified for better (or worse) results.

Fig. 5 shows the deep view of a convolution, the primary component of CNN architecture that plays a vital role in feature selection from the image. Eq. (4) was used to calculate the convolution.

$$net(i, j) = (\xi * \varphi)[i, j] = \sum_m \sum_n \xi[m, n] \varphi[i - m, j - n] \quad (4)$$

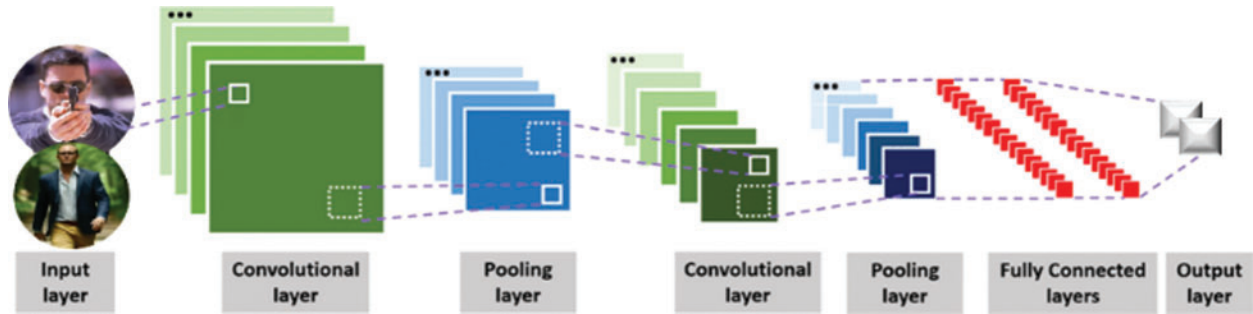


Figure 4: CNN model for the proposed intelligent detection & classification (IADC) system

where $net(i, j)$ represents the output image, ξ is the input image, φ is the kernel or filter matrix, and $*$ is the convolution.

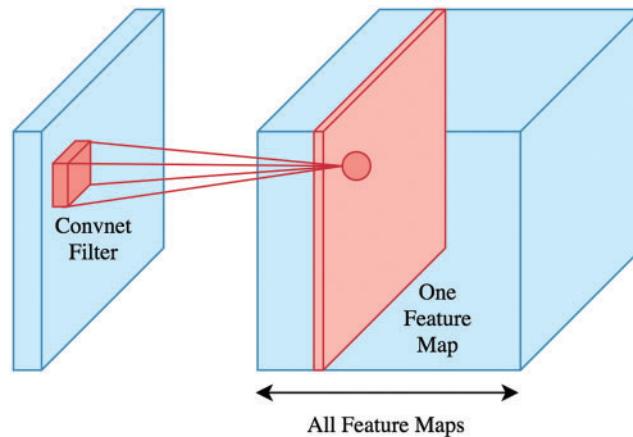


Figure 5: Convolutional layer in the CNN model

The core building block of CNN is a Convolutional layer, which has been used for feature detection.

Let's assume the image size is a 7×7 matrix with RGB channels. A kernel or feature detector or window of size 5×5 with three (R, G, B) channels and stride 1 is being used to scan the kernel over the image.

In Fig. 6, if the kernel 3×3 moves over the image 7×7 matrix having stride one. Then, the dimension of the output feature map can be calculated by Eq. (5).

$$Out = \frac{(W - F + 2P)}{S} + 1 \tag{5}$$

where, W is the image size, F is the kernel size, P is padding, and S is stride.

$$out = \frac{(7 - 3 + 2 \times 0)}{1} + 1 \Rightarrow 5$$

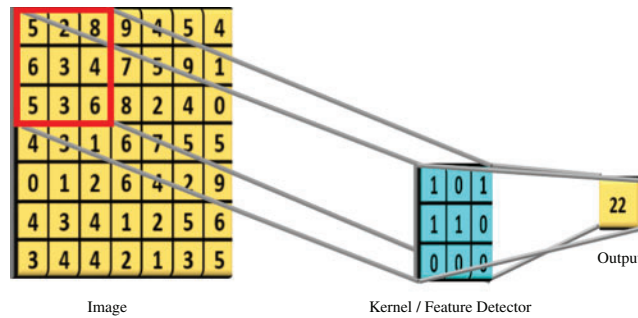


Figure 6: Producing feature mapping results with 3×3 kernel size

Therefore, the dimension of the output feature map is 5×5 . The Rectified Linear Unit (ReLU) is an activation function, which commonly uses CNN. The ReLU function’s problem is that it is not differentiable at the origin; therefore, it is difficult to use with backpropagation training.

It is defined mathematically, as shown in Eqs. (6), (7).

$$y = \max(0, x) \tag{6}$$

$$\frac{d}{dx} ReLU(x) = \begin{cases} One & \text{if } x > 0; \\ 0 & \text{otherwise} \end{cases} \tag{7}$$

Fig. 7 shows the graphic representation of ReLU, which creates a liner for all positive values of x , and $y = 0$ for all negative values of x . This activation function always produces positive and zero outputs; if the output is negative, it converts to zero.

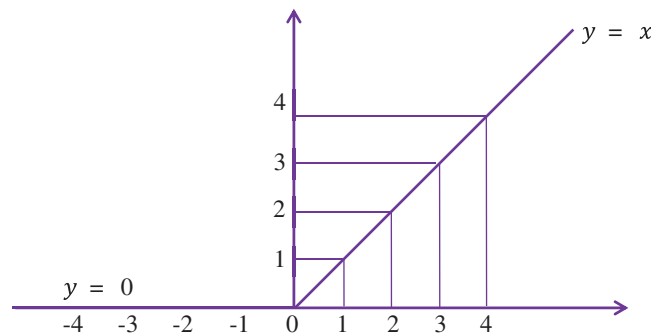


Figure 7: ReLU of the proposed IADC system using a convolutional neural network

After the convolutional layers, the max-pooling layer was used to reduce the input stream’s spatial dimension. The height and weight of the images were reduced. As shown in Fig. 8, the CNN layer of the IADC system used max-pooling with 2×2 filter size and stride 2.

Finally, a fully connected layer becomes the input for the SoftMax layer and produces the classification layer results.

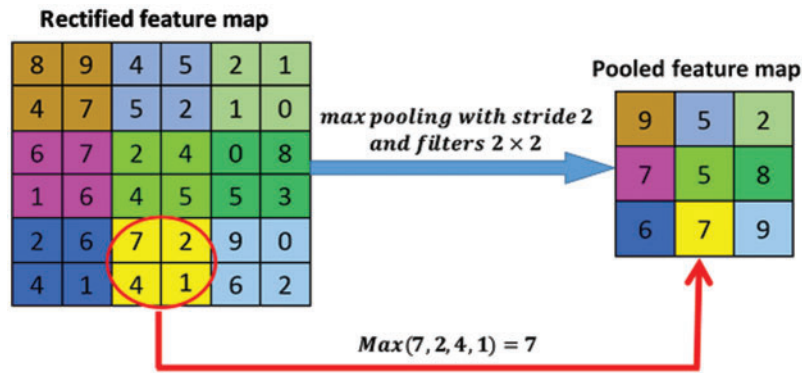


Figure 8: Max-pooling processing with filters 2×2 and stride 2

3.5 Mathematical Model of CNN Loss

$$Y = \begin{bmatrix} Y_1 \\ Y_2 \\ Y_3 \\ \vdots \\ Y_i \end{bmatrix}, \quad y = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \\ y_i \end{bmatrix}$$

where the y and Y vectors represent the estimated values from the convolutional layer and initial results. The difference between y_i and Y_i is called loss, which may be calculated by different methods. The loss is used in backpropagation to update the weights. Because the calculated values of y_i are required near the original output Y_i .

In the mathematical model, the aim is to backpropagate by taking the derivative of Eq. (8) related to weights or filter $\frac{\partial L}{\partial W}$ and $\frac{\partial L}{\partial b}$ bias. It is the cross-entropy loss that is used for classification. Because the new weights and bias will be obtained using a decent gradient algorithm by $W_{new} = W_{old} - \alpha \frac{\partial L}{\partial W}$ and $b_{new} = b - \alpha \frac{\partial L}{\partial b}$, α is the learning rate parameter.

$$L = - \sum_{i=1}^c Y_i \log(y_i) \tag{8}$$

where c = the number of classes depending upon the implementation.

we have SoftMax conversion as in Eq. (9)

$$y_i = \frac{e^{Z_i}}{\sum_{k=1}^n e^{Z_k}} \tag{9}$$

where Z_i represents logits or output units, and logits will convert into probabilities via the SoftMax conversion.

$$Z_l = \sum_{j=1}^{n_{out}} (W_{jl} * X_j) \quad (10)$$

Z_i is attained by interrelated weights with the X_j . Here, we get loss related to weights contingent on dual summations in Eq. (11). One from $j=1$ to n_{out} and the other one from $l=1$ to c . Finally, the product of two derivatives will be taken

$$\frac{\partial L}{\partial W_{jl}} = \sum_{j=1}^{n_{out}} \sum_{l=1}^c \left(\frac{\partial L}{\partial Z_l} \times \frac{\partial Z_l}{\partial W_{jl}} \right) \quad (11)$$

where $\frac{\partial y_i}{\partial Z_l}$ is the *SoftMax* derivative.

In Eq. (8), loss considering y_i as its parameter is obliquely associated to Z_i in connection with subsequent Eq. (12).

$$y_i = \frac{e^{Z_i}}{\sum_{k=1}^c e^{Z_k}} \quad (12)$$

$Z_l = \sum_{j=1}^{n_{out}} (W_{jl} * X_j)$ is given as $Z_i = Z_l$

Two demonstrations are vital, where demonstration 1, $i=l$, and demonstration 2 $i \neq l$ when $i=l$ th unit. l is the unique neuron and pivot in SoftMax output neurons; and l neuron has the highest values, and the rest are close to zero.

Case 1 ($i=l$): Now derivative of Eq. (9) via the quotient rules with reference to Z_l

$$\frac{\partial y_i}{\partial Z_{(i=l)}} = \frac{e^{Z_i} \sum_{k=1}^c e^{Z_k} - e^{Z_i} e^{Z_l}}{\sum_{k=1}^c e^{Z_k} \times \sum_{k=1}^c e^{Z_k}}$$

Taking common $\frac{e^{Z_i}}{\sum_{k=1}^c e^{Z_k}}$, we get the following:

$$\frac{\partial y_i}{\partial Z_l} = \frac{e^{Z_i}}{\sum_{k=1}^c e^{Z_k}} \left(\frac{\sum_{k=1}^c e^{Z_k} - e^{Z_l}}{\sum_{k=1}^c e^{Z_k}} \right)$$

By dividing, we get

$$\frac{\partial y_i}{\partial Z_l} = \frac{e^{Z_i}}{\sum_{k=1}^c e^{Z_k}} \left(1 - \frac{e^{Z_l}}{\sum_{k=1}^c e^{Z_k}} \right) \quad \{ \cdot \cdot \ i=l \}$$

Because we know that $y_i = \frac{e^{Z_i}}{\sum_{k=1}^c e^{Z_k}}$, the above equation can be written as

$$\frac{\partial y_i}{\partial Z_l} = y_i(1 - y_i) = y_l(1 - y_l) \quad \text{for } (i=l) \quad (13)$$

When $i \neq l$ th unit, then it has a low probability but when l is the single neuron pivot in SoftMax output neuron. Therefore,

Case 2 ($i \neq l$): Now derivative of Eq. (12) via quotient rules with reference to Z_l

$$\frac{\partial y_i}{\partial Z_l} = \frac{\frac{\partial}{\partial Z_l} e^{Z_i} * \sum_{k=1}^c e^{Z_k} - e^{Z_l} \frac{\partial}{\partial Z_l} (\sum_{k=1}^c e^{Z_k})}{\sum_{k=1}^c e^{Z_k} * \sum_{k=1}^c e^{Z_k}}$$

It can be written as

$$\frac{\partial y_i}{\partial Z_l} = 0 - \frac{e^{Z_i} * e^{Z_l}}{\sum_{k=1}^c e^{Z_k} * \sum_{k=1}^c e^{Z_k}} = - \frac{e^{Z_i}}{\sum_{k=1}^c e^{Z_k}} * \frac{e^{Z_l}}{\sum_{k=1}^c e^{Z_k}}$$

Because we know that $y_i = \frac{e^{Z_i}}{\sum_{k=1}^c e^{Z_k}}$ and $y_l = \frac{e^{Z_l}}{\sum_{k=1}^c e^{Z_k}}$, we can drive the equation as

$$\frac{\partial y_i}{\partial Z_l} = -y_i y_l \quad \text{for } (i \neq l) \tag{14}$$

We can summarize Eqs. (13), (14).

$$\frac{\partial y_i}{\partial Z_l} = \begin{bmatrix} y_i(1 - y_i) \\ -y_i y_l \end{bmatrix} \tag{15}$$

Because cross-entropy has no component of Z_l , the partial derivative of Z_l related to $\log(y_k)$ will be observed

$$L = - \sum_{i=1}^c (Y_i * \log(y_i))$$

Taking the derivative cross-entropy loss, the equation becomes

$$\begin{aligned} \frac{\partial L}{\partial Z_l} &= - \sum_{i=1}^c \left(Y_K * \frac{\partial}{\partial Z_l} \log(y_k) \right) \\ \frac{\partial L}{\partial Z_l} &= - \sum_{i=1}^c Y_K \left(\frac{\partial}{\partial y_k} \log(y_k) \right) \frac{\partial y_k}{\partial Z_l} \\ \frac{\partial L}{\partial Z_l} &= - \sum_{i=1}^c \frac{Y_K}{y_k} \frac{\partial y_k}{\partial Z_l} \end{aligned} \tag{16}$$

$\frac{\partial y_k}{\partial Z_l}$ was earlier computed for the SoftMax gradient. Two demonstrations are there $i \neq l$,

$k \neq l$ as in Eq. (15) and Eq. (16) will be divided into two parts $\frac{\partial L}{\partial Z_l} = -\frac{Y_K}{y_k} * y_k(1 - y_l) -$

$$\sum_{k \neq l}^c \left(-\frac{Y_K}{y_k} * y_k y_l \right) \text{ We can simplify this as}$$

$$\frac{\partial L}{\partial Z_l} = -Y_K(1 - y_l) + \sum_{k \neq l}^c Y_K y_l$$

We can further simplify this as

$$\frac{\partial L}{\partial Z_l} = -Y_K + Y_K y_l + \sum_{k \neq l}^c Y_K y_l$$

$$\frac{\partial L}{\partial Z_l} = y_l \left(Y_K + \sum_{k \neq l}^c Y_K \right) - Y_K$$

where $\left(Y_K + \sum_{k \neq l}^c Y_K \right)$ represents 1

$$\frac{\partial L}{\partial Z_l} = y_l - Y_K$$

$$\frac{\partial L}{\partial Z_l} = y_l - Y_l \quad \{ \cdot : k = l \}$$

Now, taking the derivative of Eq. (10), $\frac{\partial Z_l}{\partial W_{jl}}$ we get

$$\frac{\partial Z_l}{\partial W_{jl}} = x_j$$

Now, putting the values of $\frac{\partial L}{\partial Z_l}$ and $\frac{\partial Z_l}{\partial W_{jl}}$ in Eq. (11),

$$\frac{\partial L}{\partial W_{jl}} = \sum_{j=1}^{n_{out}} \sum_{l=1}^c \left(\frac{\partial L}{\partial Z_l} \times \frac{\partial Z_l}{\partial W_{jl}} \right)$$

Therefore,

$$\frac{\partial L}{\partial W_{jl}} = \sum_{j=1}^{n_{out}} \sum_{l=1}^c (y_l - Y_l) x_j \tag{17}$$

Eq. (17) is the derivative of loss related to weights for the fully connected layer. Once $\frac{\partial L}{\partial W_{jl}}$ is obtained by applying gradient descent on the fully connected layer; the updated weights will be achieved.

4 Simulation and Results

MATLAB was used to simulate the proposed IADC system using a CNN. The dataset used for the simulation contained 920 images showing persons with and without weapons. The dataset was further divided into training (700) and validation (220). Figs. 9 and 10 show the accuracy and extent of loss in the proposed system. Training accuracy achieved a level of 99.41%; validation accuracy was 96.74%.

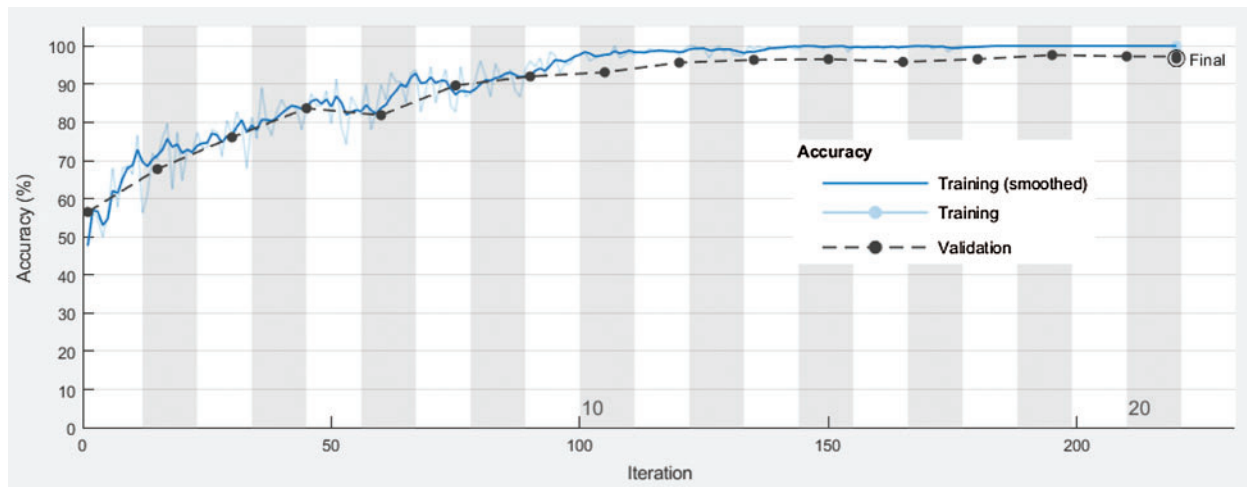


Figure 9: Performance analysis of the proposed IADC System using CNN related to iterations and accuracy

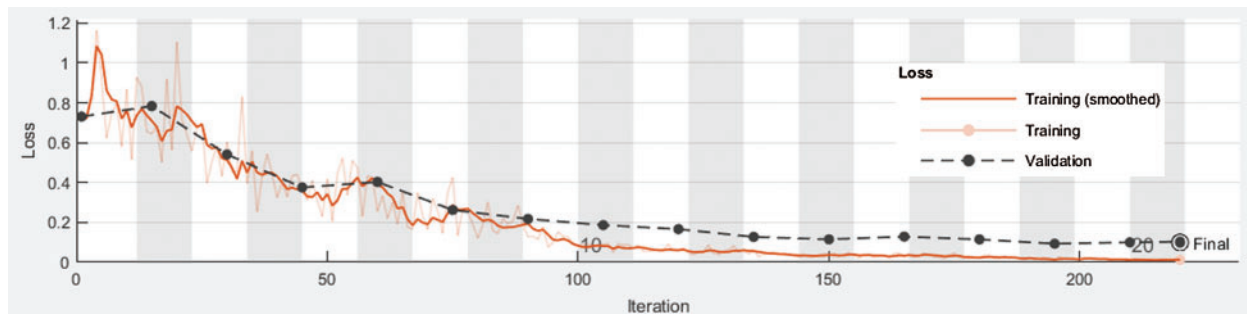


Figure 10: Performance analysis of the proposed IADC system using CNN related to iteration vs. loss

Fig. 10 shows the training and validation performance of the proposed IADC system related to iteration and loss. We observed that the proposed system resulted in loss rates of 0.01 during training and 0.09 during validation.

Fig. 11 shows the randomly selected labeled output images based on the proposed system. The results showed that the proposed IDAC system classified persons into two classes: With ammunition and without ammunition.



Figure 11: Randomly selected sample images from the dataset

Tab. 1 compares the performance of the proposed IADC system with previously published models. The results show that, of the previously published models, the OverFeat-3 system [5] offered the highest precision with 93% during training and 89% during testing. In comparison, the proposed IADC system operated at 99.41% during training and 96.74 % during testing. Moreover, the IADC system provided more accurate results than previously published methods like VGGNet, OverFeat-1, OverFeat-2, and OverFeat-3 [5].

Table 1: Comparison of the IADC system using CNN with previously published models

Model	Train Precision (%)	Test Precision (%)
VGGNet [5]	57	46
OverFeat-1 [5]	62	56
OverFeat-2 [5]	69	64
OverFeat-3 [5]	93	89
Proposed IADC system using CNN	99.41	96.74

5 Conclusions and Future Studies

In conclusion, the Convolutional Neural Network application in security systems offers more precise detection of armed persons and weapons. Additionally, it has given more accurate results than previously published methods such as VGGNet, Overfeat-1, Overfeat-2, and Overfeat-3. The proposed IADC system achieved a 96.74% accuracy rate and a 3.26% loss rate.

In future studies, the Yolo model can be used to obtain more precise results through comparing results. Moreover, improved precision in detecting and classifying a wider variety of weapons and ammunition can be achieved.

Acknowledgement: Thanks to our families and colleagues, who provided moral support.

Funding Statement: The authors received no specific funding for this study.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding this study.

References

- [1] M. Grega, A. Matiolański, P. Guzik and M. Leszczuk, "Automated detection of firearms and knives in a CCTV image," *Sensors*, vol. 16, no. 47, pp. 1–16, 2016.
- [2] R. Olmos, S. Tabik and F. Herrera, "Automatic handgun detection alarm in videos using deep learning," *Neurocomputing*, vol. 275, pp. 66–72, 2018.
- [3] A. Atta, M. A. Khan, S. Abbas, G. Ahmad and A. Fatimat, "Modelling smart road traffic congestion control system using machine learning techniques," *Neural Network World*, vol. 29, no. 2, pp. 99–110, 2019.
- [4] Ahmed, "Detection of objects in images using region-based convolutional neural networks," *United States Patent*, vol. 2, pp. 1–20, 2018.
- [5] J. Lai and S. Maples, "Developing a real-time gun detection classifier," *Technical Report*, vol. 2017, pp. 1–4, 2017.
- [6] A. Caragliu and C. F. D. Bo, "Smart innovative cities: The impact of smart city policies on urban innovation," *Technological Forecasting and Social Change*, vol. 142, no. December 2017, pp. 373–383, 2019.
- [7] N. Cowan, "The many faces of working memory and short-term storage," *Psychonomic Bulletin & Review*, vol. 24, no. 4, pp. 1158–1170, 2017.
- [8] A. Ahi and A. V. Singh, "Role of distributed ledger technology (DLT) to enhance resiliency in internet of things (IOT) ecosystem," in *Amity Int. Conf. on Artificial Intelligence*, Dubai, United Arab Emirates, pp. 782–786, 2019.
- [9] Almotaeryi and Resheed, "An integrated framework for CCTV infrastructures deployment in KSA: Towards an automated surveillance," *Doctoral Thesis*, North Umbria University, pp. 1–183, 2018.
- [10] L. Perez and J. Wang, "The effectiveness of data augmentation in image classification using deep learning," *Arxiv Preprint Arxiv*, vol. 2017, pp. 1–8, 2017.
- [11] M. I. Razzak, S. Naz and A. Zaib, "Deep learning for medical image processing: Overview, challenges and the future," *Lecture Notes in Computational Vision and Biomechanics*, vol. 26, pp. 323–350, 2018.
- [12] O. Sheremet, K. Sheremet, O. Sadovoi and Y. Sokhina, "Convolutional neural networks for image denoising in infocommunication systems," in *Int. Scientific-Practical Conf. Problems of Infocommunications Science and Technology*, Kharkiv, Ukraine, pp. 429–432, 2019.
- [13] K. Zhang, W. Zuo, Y. Chen, D. Meng and L. Zhang, "Beyond a Gaussian denoiser: Residual learning of deep CNN for image denoising," *IEEE Transactions on Image Processing*, vol. 26, no. 7, pp. 3142–3155, 2017.

- [14] Y. J. Cha, W. Choi and O. Büyüköztürk, “Deep learning-based crack damage detection using convolutional neural networks,” *Computer-Aided Civil and Infrastructure Engineering*, vol. 32, no. 5, pp. 361–378, 2017.
- [15] Y. Yang, P. Bi and Y. Liu, “License plate image super-resolution based on convolutional neural network,” in *IEEE 3rd Int. Conf. on Image, Vision and Computing*, Chongqing, China, pp. 723–727, 2018.
- [16] P. Tsoutsas, P. Fitsilis, L. Anthopoulos and O. Ragos, “Nexus services in smart city ecosystems,” *Journal of the Knowledge Economy*, vol. 11, no. 3, pp. 1–20, 2020.
- [17] A. Handa, P. Garg and V. Khare, “Masked neural style transfer using convolutional neural networks,” in *Int. Conf. on Recent Innovations in Electrical, Electronics & Communication Engineering*, Bhubaneswar, India, pp. 2099–2104, 2018.
- [18] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” *ArXiv Preprint ArXiv*, vol. 2014, pp. 1–14, 2014.
- [19] V. Araujo, K. Mitra, S. Saguna and C. Åhlund, “Performance evaluation of fiware: A cloud-based IoT platform for smart cities,” *Journal of Parallel and Distributed Computing*, vol. 132, pp. 250–261, 2019.
- [20] R. Girshick, J. Donahue, T. Darrell, J. Malik, U. C. Berkeley *et al.*, “Deformable part models are convolutional neural networks,” in *IEEE Conf. on Computer Vision and Pattern Recognition*, Boston, USA, pp. 437–446, 2015.
- [21] H. Jiang and E. L. Miller, “Face detection with the faster R-CNN,” in *12th IEEE Int. Conf. on Automatic Face & Gesture Recognition*, Washington, USA, pp. 650–657, 2017.
- [22] Y. Hua, R. Li, Z. Zhao, X. Chen and H. Zhang, “Gan-powered deep distributional reinforcement learning for resource management in network slicing,” *IEEE Journal on Selected Areas in Communications*, vol. 38, no. 2, pp. 334–349, 2020.
- [23] A. Atta, S. Abbas, M. A. Khan, G. Ahmed and U. Farooq, “An adaptive approach: Smart traffic congestion control system,” *Journal of King Saud University-Computer and Information Sciences*, vol. 32, no. 9, pp. 1012–1019, 2018.
- [24] M. Asadullah, M. A. Khan, S. Abbas, A. Athar, S. S. Raza *et al.*, “Blind channel and data estimation using fuzzy logic-empowered opposite learning-based mutant particle swarm optimization,” *Computational Intelligence and Neuroscience*, vol. 2018, no. 5, pp. 1–12, 2018.
- [25] K. Iqbal, M. A. Khan, S. Abbas, Z. Hasan and A. Fatima, “Intelligent transportation system (ITS) for smart-cities using mamdani fuzzy inference system,” *International Journal of Advanced Computer Science and Applications*, vol. 9, no. 2, pp. 94–105, 2018.
- [26] G. Ahmad, M. A. Khan, S. Abbas, A. Athar, B. S. Khan *et al.*, “Automated diagnosis of hepatitis B using multilayer mamdani fuzzy inference system,” *Journal of healthcare engineering*, vol. 2019, no. 1, pp. 1–11, 2019.
- [27] S. Zahra, M. A. Khan, M. N. Ali and S. Abbas, “Standardization of cloud security using mamdani fuzzifier,” *International Journal of Advanced Computer Science and Applications*, vol. 9, no. 3, pp. 292–297, 2018.
- [28] Gu and Jingtian, “The application of convolutional neural network in security code recognition,” *Journal of Physics: Conference Series*, vol. 1187, no. 4, pp. 42064, 2019.
- [29] Q. Chen, X. Gan, W. Huang, J. Feng and H. Shim, “Road damage detection and classification using mask R-CNN with densenet backbone,” *Computers, Materials & Continua*, vol. 65, no. 3, pp. 2201–2215, 2020.
- [30] R. Chen, L. Pan, C. Li, Y. Zhou, A. Chen *et al.*, “An improved deep fusion CNN for image recognition,” *Computers Materials & Continua*, vol. 65, no. 2, pp. 1691–1706, 2020.
- [31] Y. Chen, X. Qin, L. Zhang and B. Yi, “A novel method of heart failure prediction based on dpcnnxgboost model,” *Computers Materials & Continua*, vol. 65, no. 1, pp. 495–510, 2020.
- [32] Y. Zhao, J. Cheng, P. Zhang and X. Peng, “ECG classification using deep CNN improved by wavelet transform,” *Computers Materials & Continua*, vol. 64, no. 3, pp. 1615–1628, 2020.