APRIL 2022 M.Sc. in Electrical and Electronics Engineering MUSTAFA HUSSAIN RAFEEQ

REPUBLIC OF TURKEY

GAZİANTEP UNIVERSITY

GRADUATE SCHOOL OF NATURAL & APPLIED SCIENCES

A REAL-TIME FALL DETECTION OF ELDERLY PEOPLE IN INDOOR ENVIRONMENTS

M.Sc. THESIS

IN

ELECTRICAL AND ELECTRONICS ENGINEERING

BY

MUSTAFA HUSSAIN RAFEEQ

APRIL 2022

A REAL-TIME FALL DETECTION OF ELDERLY PEOPLE IN INDOOR ENVIRONMENTS

M.Sc. Thesis

in

Electrical and Electronics Engineering

Gaziantep University

Supervisor

Asst. Prof. Dr. Serkan ÖZBAY

by

Mustafa Hussain RAFEEQ

April 2022

A REAL-TIME FALL DETECTION OF ELDERLY PEOPLE IN INDOOR ENVIRONMENTS

submitted by **Mustafa Hussain RAFEEQ** in partial fulfillment of the requirements for the degree of Master of Science in **Electrical and Electronics Engineering, Gaziantep University** is approved by,

Prof. Dr. Mehmet İshak YÜCE
Director of the Graduate School of Natural and Applied Sciences …………..

Prof. Dr. Ergun ERÇELEBİ
Head of the Department of Electrical and Electronics Engineering …………..

Asst. Prof. Dr. Serkan ÖZBAY
Supervisor, Electrical and Electronics Engineering …………..

Exam Date: 21 April 2022

Examining Committee Members:

Asst. Prof. Dr. Serkan ÖZBAY
Thesis Supervisor, Electrical and Electronics Engineering
Gaziantep University …………..

Prof. Dr. Ergun ERÇELEBİ
Electrical and Electronics Engineering
Gaziantep University …...............

Asst. Prof. Dr. Mehmet ARICI
Electrical and Electronics Engineering
Gaziantep Islam Science and Technology University …...............

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I havefully cited and referenced all material and results that are not original to this work.

Mustafa Hussain RAFEEQ

ABSTRACT

A REAL-TIME FALL DETECTION OF ELDERLY PEOPLE IN INDOOR ENVIRONMENTS

RAFEEQ, Mustafa Hussain
M.Sc. in Electrical and Electronics Engineering
Supervisor: Asst. Prof. Dr. Serkan ÖZBAY
April 2022
56 pages

The extraordinary effectiveness of existing image-based AI technologies and the resulting interest in their use in key decision-making processes have sparked a rise in attempts to make such intelligent systems visible and understandable. The technology we use today is the result of several researchers achieving artificial intelligence achievements. Artificial intelligence leverages computers and machines to mimic the human mind's problem-solving and decision-making capabilities. For example, some researchers have demonstrated that convolutional neural networks with shorter connections between layers adjacent to the input and those close to the output can be significantly deeper, more accurate, and efficient to train.

This study serves as a framework for distinguishing fall event from other indoor natural human activities. In this study, a modified DenseNet 121 approach that relies on the deletion of some layers and the addition of others for fall detection is presented. With the proposed approach, losses are reduced, and network variables like weights and learning rate are updated using the binary cross-entropy loss function. Finally, the sigmoid classifier identifies human falls using binary classification. It is shown that the proposed system achieves a precision of 98.83% on experiments, outperforming existing modern models.

Key Words: Convolutional Neural Networks, Deep Learning, Fall Detection.

ÖZET

KAPALI ORTAMLARDA BULUNAN YAŞLILAR İÇİN GERÇEK ZAMANLI DÜŞME ALGILAMA SİSTEMİ

RAFEEQ, Mustafa Hussain
Yüksek Lisans Tezi, Elektrik ve Elektronik Mühendisliği
Danışman: Dr. Öğr. Üyesi Serkan ÖZBAY
Nisan 2022
56 sayfa

Mevcut görüntü tabanlı yapay zeka teknolojilerinin olağanüstü etkinliği ve temel karar verme süreçlerinde kullanımlarına duyulan ilgi, bu tür akıllı sistemleri görünür ve anlaşılır hale getirme girişimlerinde bir artışa yol açtı. Bugün kullandığımız teknoloji, bazı araştırmacıların yapay zeka başarıları elde etmelerinin sonucudur. Yapay zeka, insan zihninin problem çözme ve karar verme yeteneklerini taklit etmek için bilgisayarlardan ve makinelerden yararlanır. Örneğin, bazı araştırmacılar girdiye bitişik katmanlar ile çıktıya yakın katmanlar arasında daha kısa bağlantılara sahip konvolüsyonel sinir ağlarının önemli ölçüde daha derin, daha doğru ve eğitilmesinin verimli olabileceğini göstermiştir.

Bu çalışma, düşme olayını diğer iç mekan doğal insan aktivitelerinden ayırt etmek için bir çerçeve görevi görmektedir. Bu çalışmada, düşme tespiti için bazı katmanların silinmesine ve diğerlerinin eklenmesine dayanan değiştirilmiş bir DenseNet 121 yaklaşımı sunulmaktadır. Önerilen yaklaşımla kayıplar azaltılır ve ikili çapraz entropi kaybı fonksiyonu kullanılarak ağırlıklar ve öğrenme oranı gibi ağ değişkenleri güncellenir. Son olarak, sigmoid sınıflandırıcı, ikili sınıflandırmayı kullanarak insan düşüşlerini tanımlar. Önerilen sistemin, mevcut modern modellerden daha iyi performans göstererek, deneysel çalışmalarda %98.83 hassasiyete ulaştığı gösterilmiştir.

Anahtar Kelimeler: Konvolüsyonel Sinir Ağları, Derin Öğrenme, Düşme Algılama.

"Dedicated to my family"

ACKNOWLEDGEMENTS

I would like to thank my supervisor, Asst. Prof. Dr. Serkan ÖZBAY for his guidance and support throughout the study. I am thankful for his encouragement and motivation.

I would like to express my love and gratitude to my family for their support, always best wishes. Special thanks to my wife for her support and encouragement. I present this thesis on the soul of my father and mother, and they passed away while I was studying for my M.Sc.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

CNN	Convolutional Neural Network
RNN	Recurrent Neural Network
ReLU	Rectifier Linear Unit
GPU	Graphic Processing Unit
FCN	Fully Convolutional Network
AI	Artificial Intelligence
ANN	Artificial Neural Network

CHAPTER I
INTRODUCTION

1.1 Overview

A fall is an incidence in which an individual falls from a standing situation while incapable of helping oneself.

Elder people, particularly those who live solely and without aid, suffer from improbable falls due to the fragility of older people's bodies, and besides, likely fall dangers (slippery floors, closed lighting, improper furnishing, crossed paths, etc.). That is why a humanoid fall detection method is very mandatory for several people.

Recently, particular nations have seen a substantial increment in their older people according to fall detection. Falls are a fundamental cause of harmful mortality in people over 79 [1]. According to the National Institutes of Health Investigations in the United States, 1.7 million older people in the US suffer from fall-related casualties every year [2].

Besides that, China suffers from the top community aging in Earth date, with the figure of nationals anticipated to scale to about 35% (2020 – 2050) [2]. Moreover, according to research [3], 92% of the elderly lives alone in their home environment, with 29% living alone in a house. In reality, if they do not have either physical detriments, it has been certain that almost half of the elderly persons who are laying on the floor for more than a timecausala fall would pass away within six months [4].

Falls are a further regular occurrence among the elderly, and they can have earnest health effects. According to studies, 28-34% of older persons fall at the lowest once a year [5]. Additionally, for more than 87% of older persons, falling is the second leading to arbitrary death [6]. Falls are among the many common causes of damage in older persons, including fractures, brain damage, and strained muscles [7].

An disabled person may repose on the floor for an overextended period before receiving help [8]. Although fall observation and modification systems cannot avoid falls, they can assist in lessening problems by guaranteeing that the falling perso.

receives support as soon as possible [8]. Much effort has gone into evolving automated human fall detection systems [9]. There has been a steady augment in research on video surveillance systems for disposition human behavior [6]. Computer vision systems are a new encouraging option that can aid older people to stay at home longer and improve their benchmark of life by providing a protected environment [10].

Falling is considered one of the most common reasons for death in the elderly [7]. Consequently, it is sensitive to carry out a penetration observation system for older people to automatically identify fall conducts inside the range, advising care givers.

Unfortunately, most fall detection devices can't distinguish between common and actual falls. As a result, this study aims to search for the best techniques for inspection falls by analyzing the movement and structure of the human body. The study will utilize a webcam to automatically watch older folks in their homes to identify human falls. This research proposes a deep learning and vision-based system for human fall observation and categorization, monitoring older people in their homes. We adjusted a transfer learning Densenet 121 model to detect person fall conduct. We also put our suggested model to the test against the UR fall detection dataset which is one of the most famous dataset. Our proposed model performs brilliantly compared to other contemporary models on these datasets.

1.2 Aims and Objectives

This study aims to use camera sensor data to identify human falls automatically. The research is established on a lowliest-cost, off-the-shelf video camera surveillance system to help older people remain alone in their homes. Also, this study aims to look into the best approaches for detecting and characterizing occupant falls in a residential context. A system for observing human activities at-home setting and revealing a fall accident based on movement data and changes in the direction of a person's shape is provided in this study. The other goals have been defined to help attain this target:

1. To conduct a thorough review of the literature on existing technology for monitoring and detecting falls in older adults.
2. To apply processing algorithms to capture things captured with a lowest-cost camera, leading in outline identification and feature removal origin to determine the objects' physical shape, situation, and exercise.

3. To use the techniques to create a simulated home scenario in which to recognize, track, and identify a person's physical movement and position.
4. To research and develop methods for analyzing the results of 3. To recognize problem scenarios/situations by automatic classification of situations.
5. To use processed camera footage, evaluate the temporal link amidst a concatenation of frames (low rate frames) to determine oldsters' existence and behavior.

1.3 Problem Definition

This project aims to modernize elderly care with modern technology to execute a fall detection system that could potentially be utilized as a subsystem for helping the elderly, support personnel, and other persons. The following features would have to be included in such a system:
1. The system should operate in real-time. That is, it should detect a fall as soon as it occurs. The fall detection algorithm, in particular, must run at a faster rate than the input signal, which is the video stream.
2. The system should detect a fall from a distance of up to six meters, which is feasible from aresident's standpoint, as a standard room is roughly 24 square meters or 6x4 meters.
3. The users of the systems should not feel as if their privacy is being invaded; hence little intrusiveness is desired.
4. Interoperability, as in the system, should provide a confidence value indicating how certain a fall has been identified. Other techniques, such as an alarm system, could use this information to listen for occurrences and make judgments depending on the confidence value.

1.4 Contents of the Thesis

The remaining chapters of this thesis are divided into four sections. Background about deep learning and CNN predefined architectures and fall systems classification technologies are covered in the second chapter's literature study. The Materials and Methods section was addressed in the third chapter. The research methodologies utilized in this study will be explained in the order in which they were used: Dataset, Proposed Architecture, Data Normalization, Leaky Rectified Linear, Sigmoid Function, and Cost Function are all terms that can be used to describe a data collection. Fourth Chapter The findings of this investigation. Finally, in Chapter Five, there is a section called Conclusions and Future Work. This chapter summarizes the thesis's findings and proposes future studies monitoring older individuals' everyday activities and detecting abnormal events such as falls.

CHAPTER II

THEORY AND LITERATURE REVIEW

2.1 Theory

This section examines the theory and methods underpinning a neural network's learning capabilities in greater depth, focusing on neural networks that use supervised learning to solve the binary classification problem. First, the cost functions and how they relate to the backpropagation algorithm are checked. Besides, the description of the neural networks and their relationships in image data using convolutional operations are presented.

The terms "Artificial Intelligence," "Machine Learning," "Artificial Neural Networks," and "Deep Learning" are frequently interchanged in the software development industry to express the same or very similar concepts and ideas. So, it's no surprise that, despite their origins in different eras and stages of development, these concepts have something in common: a computer is given directives to learn and search the best (or a decent) resolution to a case, rather than instructions to solve a problem, oppositely to traditional programming, which describes specific subtasks and orders to disband a more significant problem [11].

2.1.1 Deep Learning

The rapid advancement in computation techniques plays a vital role in providing a robust framework by ANNs with deep manners for supervised learning. Deep learning is a neural network with multiple layers from nonlinear processing units. Every successive layer takes the output from the last layer as input. The network can extract the combination hierarchy features from much data by employing these layers. Deep learning can exemplify more complicated functions if the number of layers and units in a monocular layer increment. Given sufficient labeled training datasets and suitable models, deep learning approaches can aid humans in establishing mapping functions for process convenience [12].

Deep learning is a field of machine learning that learn computers to do what cames naturally to humanitarians: learn from experiment. Machine learning algorithms use computational methods to "learn" information immediately from data without depending on a pre-arranged equation as a paradigm [13]. Deep learning is particularly suited for image recognition, which is essential for solving faces detection, movement detection, and numerous advanced driver aid technologies like as independents driving, aisles detection, pedestrian detection, and independents parking. Deep learning includes four significant architectures. They are Restricted Boltzmann Machines (RBMS), Deep Belief Networks (DBNS), Autoencoder (AE), and Convolutional Neural Networks.

2.1.1.1 Convolutional Neural Networks

Convolutional Neural Networks (CNN) are the most communal category of deep learning. It was contrived by Yann LeCun and foremost applied to the recognition of handwritten digits, in which CNN stretched the commercial level in the 1990s. In modern years, in the domain of computer vision, CNN has as well played a good execution, which is fundamentally due to its application to the picture data of the particular network building [14]. It is usually utilized with high-dimensionality data like pictures and videos. The instrument of CNN working is identical to that in ANN. The essential variance is that CNN uses two-dimensional filters or weighing convolved over the picture data. CNN can be applied for equally supervised and unsupervised model learning. The supervised learning mechanism includes providing the input data with its expected output (true labels) for the system to create a prediction model. This model can map features between input and output, such as the image classification process, as shown in Figure 2.1 [15].

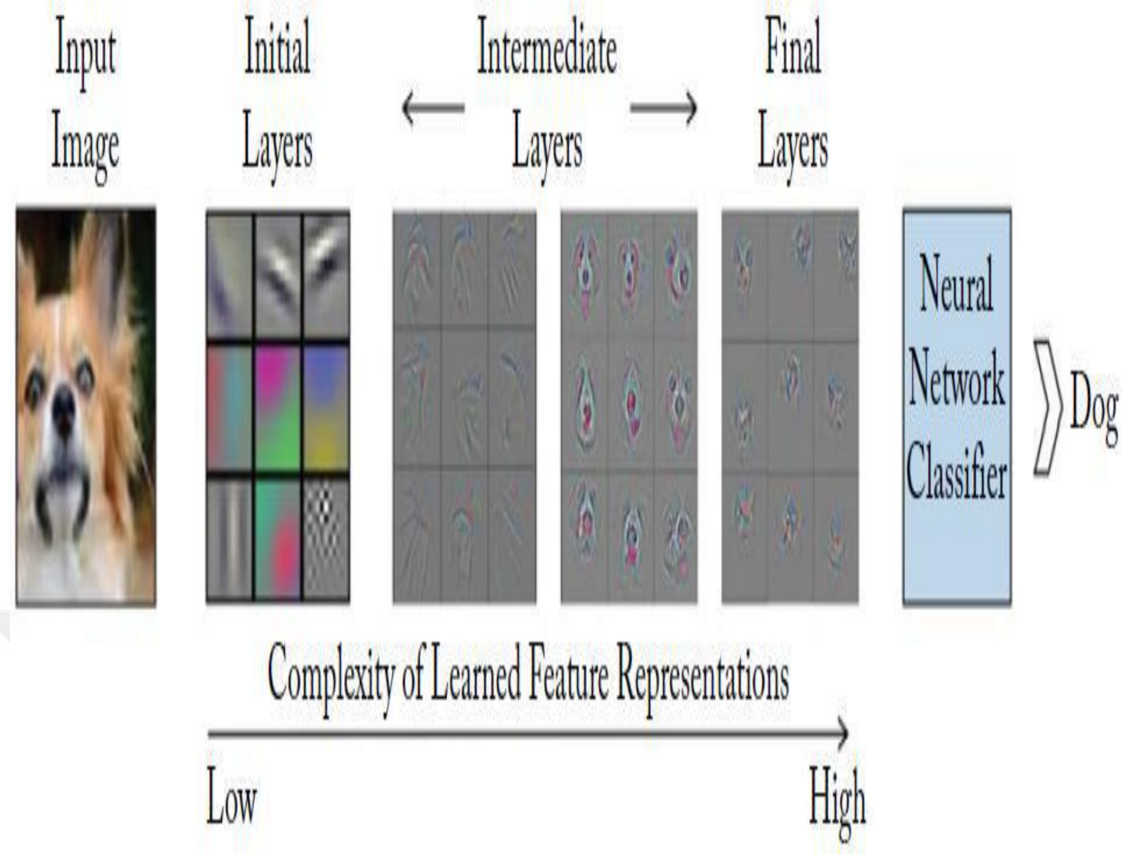

Figure 2.1 Feature Mapping from Input to Output

2.1.1.2 Convolutional Layers

The convolutional layers are the most critical layer in CNN. These layers contain a group of filters or kernels that are convolved with the input picture to extract the advantage map. These filters are weights organized in two-dimensional (in single-channel) or three-dimensional. The learning operation includes many iterations. The weights are primed randomly and updated continuously according to the expected output. The filters are convolving with the convolution layer input, as shown in Figure 2.2.

The figure illustrates the coevolution processes between a 4x4 input feature map and a 2x2 filter to generate a 3x3 output feature map. First, the filter is multiplied with a highlighted patch (also 2x2) of the input feature map and sums up all values to create a single worth in the output property map [16].

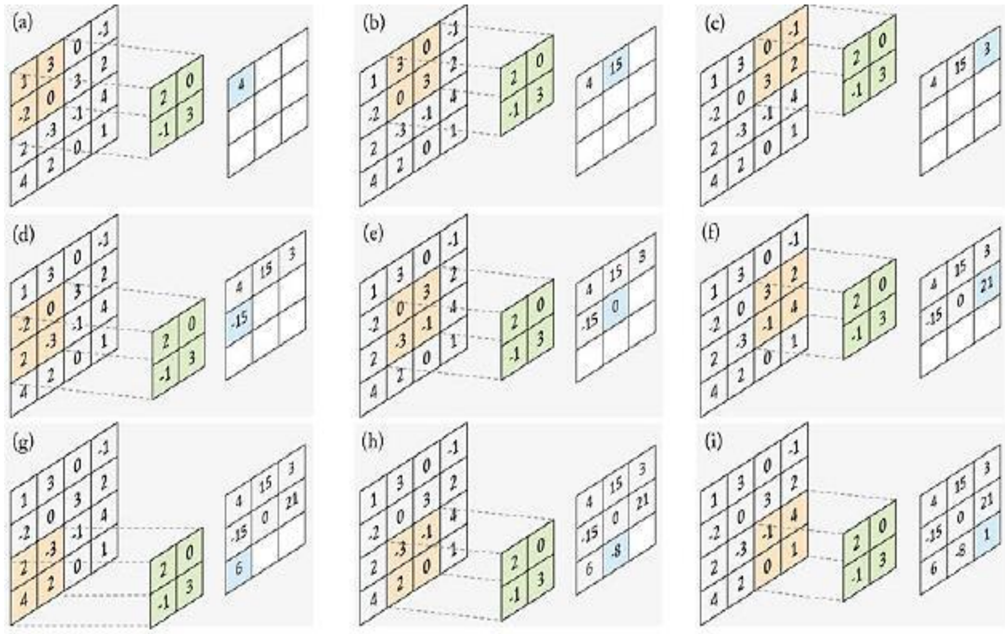

Figure 2.2 Converting an Image into a Feature Map

2.1.1.3 Fully Connected Layers

Fully connected layers can be classified as convolution layers with weights of 1x1 sizes. The units of these layers are connected with all the units of the previous layer. These layers' action is like that in ANN that comprises multiplication of the property maps inputs with the identical loads and adding the bias vector, then employing the nonlinear activation function. Fully connected layers are primarily utilized in image rating and manner recognition [17].

2.1.1.4 Pooling Layer

The input is down sampled using a pooling layer. It is usually introduced after the convolution and activation have been completed. As its name suggests, the pooling layer summaries and pools the input for an existing feature, regardless of its specific location. In other words, it can be demonstrated that pooling is unaffected by minor input translations; this suggests that even if we change the input by a small amount, the majority of the pooling output values stay the same. Max pooling is a popular method that outputs the maximum value in a 2*2 zone (or a small region in general). Average pooling is shown in Figure 2.3. L2 norm pooling and weighted average pooling based on the distance to the center pixel are all prominent pooling types. There are no parameters to learn at the pooling layer because it performs a fixed action.

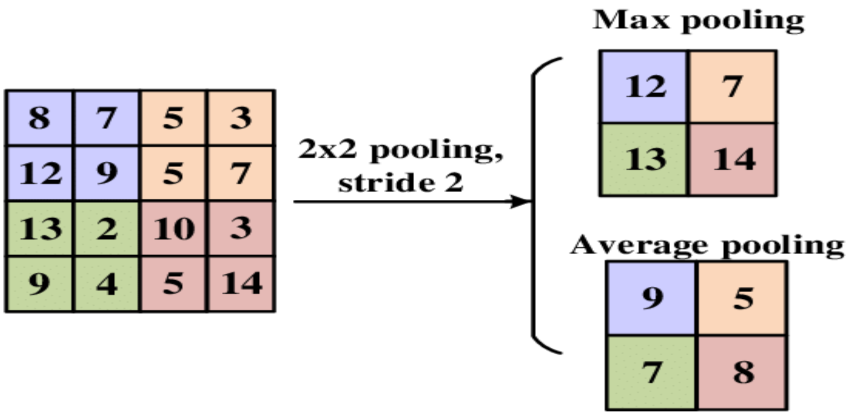

Figure 2.3 Pooling layer

2.1.1.5 Nonlinearity

A nonlinear activation or piece-wise linear function is frequently used after the weight layers in a CNN. The activation function squashes a real-valued input into a finite ambit, for example [0; 1] or [-1; 1]. A nonlinear function next the weighing layers is critical because it enables a neural network forlearn nonlinear mappings. A stacked network on the heaviness layers is tantamount to a linear plotting from input to the output field without nonlinearities.

A nonlinear function is a switching or chosen mechanism that determines if whether a neuronal will vigor or not based on whole of its inputs. The activation functions frequently utilized in deep networks are differentiable for error backpropagation. The most frequent activation functions used in deep neural networks are shown in Figure 2.4.

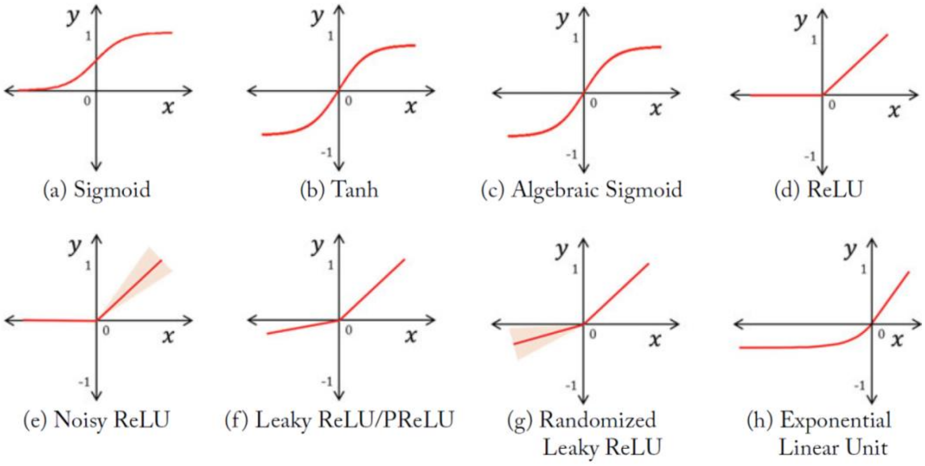

Figure 2.4 Some of the most popular activation functions utilized in deep neural networks.

2.1.1.6 Densenet

The utilize from skip and shortcut relations in architectures like ResNet, FractalNet, ResNext, and expressway networks avert the vanishing gradient trouble and enables the training of like deep networks. DenseNet expands this thought by reproduceing the output form every layer to all posterior layers, effectively facilitation the information reproducing in the fore and a back directions during network training; this permits all layers in the network to “speak” to all other automatically Figure 2.5.

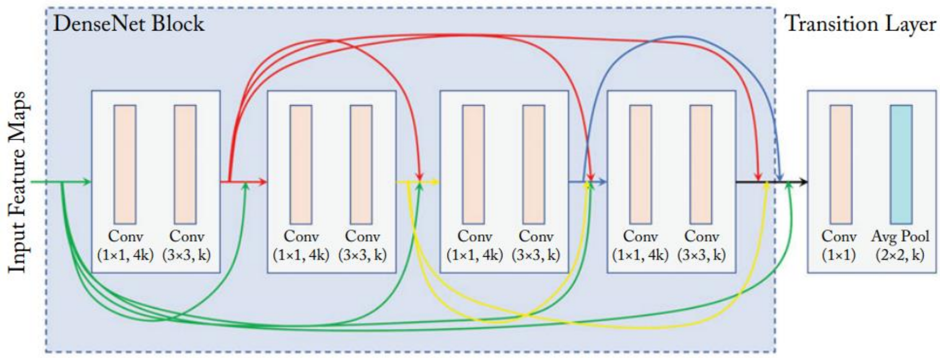

Figure 2.5 A densely connected CNN block with five layers.

Note that all layer obtains features from all forerunning layers. A convolution layer is utilized toward the finish of allbushy block to minimize the feature collection dimensionality. An epitome DenseNet is shown in Figure 2.6. Meanwhile the information influx is forward, all firstly layer is connected to the following layers. These linkage are realized by concatenating the advantage maps of a layer with the advantage maps from all the previous layers, and this is in dissimilarity to ResNet, where pass connections are utilized to add the output from an firstly layer before processing it through the following ResNet block. There are three main consequences of like forthright connections in a DenseNet. First, the features from the initial layers are forthright passed on to the later layers without any wastage of information. Second, the number of input conduits in the subsequent layers of a DenseNet growings rapidly because of the concatenation of the preceding feature maps from every firstly layers. To hold the network computations feasible, the number of output conduits (as well called the "growth average" in DenseNets) for every convolutional layer is comparatively quite low. Third, the series of feature maps can only be achieved when their spatial sizes match every other. Consequently, a densely connected CNN depends of doubled blocks with bushy engagement within the layers. The gathering or strided convolutional process is performed in amidst these blocks to collapse the input to a consolidated representation. The layers that reduce feature representations' size amidst every pair of DenseNet blocks are named the "transition layers." The transition layers are batch normalization, a 1×1 convolutional layer, and a 2×2 rate pooling layer.

Layers	Output Size	DenseNet-121($k = 32$)	DenseNet-169($k = 32$)	DenseNet-201($k = 32$)	DenseNet-161($k = 48$)
Convolution	112×112	7×7 conv, stride 2			
Pooling	56×56	3×3 max pool, stride 2			
Dense Block (1)	56×56	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 6$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 6$
Transition Layer (1)	56×56	1×1 conv			
	28×28	2×2 average pool, stride 2			
Dense Block (2)	28×28	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 12$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 12$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 12$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 12$
Transition Layer (2)	28×28	1×1 conv			
	14×14	2×2 average pool, stride 2			
Dense Block (3)	14×14	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 24$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 32$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 48$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 36$
Transition Layer (3)	14×14	1×1 conv			
	7×7	2×2 average pool, stride 2			
Dense Block (4)	7×7	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 16$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 32$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 32$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 24$
Classification Layer	1×1	7×7 global average pool			
		1000D fully-connected, softmax			

Figure 2.6 Variants of DenseNet Architectures

At every variant, the number of bushy blocks remains the same. Though, the growth average and the figure of convolution layers are incremented to styling larger architectures. All transition layer is implemented as a dimensionality lowering layer (with 1x1 convolutional candidate) and an medium pooling layer for sub-sampling. The concatenated advantage maps for the past layers in a DenseNet block are not tunable. Consequently, every layer learns its exemplification and concatenates it with the international information from the prior network phases. This information is then providing to the following layer, which can add extra information but then cannot immediately change the international information learned by the prior layers. This styling dramatically minimizes the figure of tunable parameters. It explicitly differentiates amidst the world wide situation of the network and the domestic contribution additions that all layer makes to the international situation. DenseNet borrows various resolvepicks from the priorbetter approaches. Token, a pre-activation mechanism where every convolution layer is outdistanced by a Batch Normalization and a ReLU layer. Analogously, predicament layers with 11 candidate are utilized to lessen the figure of input advantage maps before processing them meanwhile the 3x3 candidate layers. DenseNet achieves situation of the art performance on a figure of datasets like as MNIST, CIFAR10, and CIFAR100 with a comparatively less figure of parameters compared to ResNet.

2.1.2 CNN Predefined Architectures

In this subsection, CNN architecture designs using the basic building blocks are discussed. However, early architectures popular in computer vision and the most recent CNN models relatively built on the conventional methods are also discussed.

2.1.2.1 LeNet

LeNet architecture is one of the latest designs of CNNs. It is used to identify handwritten digits. It contains five weight layers; therefore, it is called the LeNet-5 model. It comprises two convolution layers and a max-pooling layer for feature extraction. Later, one convolution layer, after two connected layers across the model's finish, acts as a classifier on the removed features. Warrant that the weight layer's activations are crushed utilizing nonlinearity [18]. The program architectures for training on the MNIST digit dataset are given as shown in Figure 2.7.

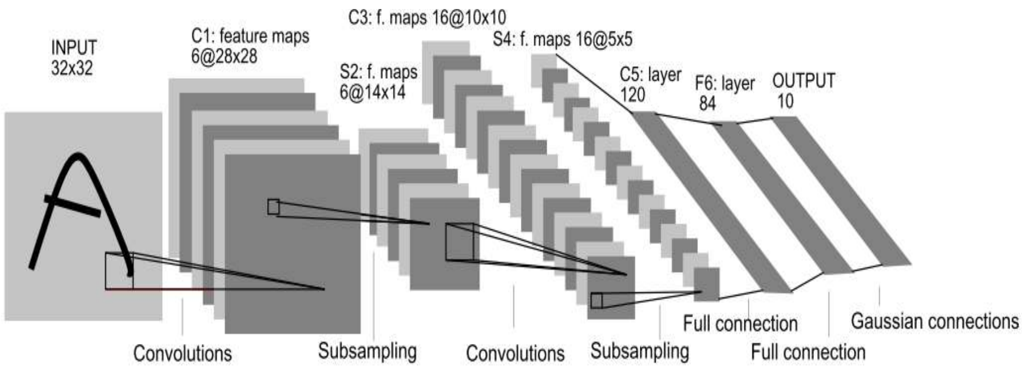

Figure 2.7 The LeNet-5 Architecture.

2.1.2.2 AlexNet

AlexNet architecture was the initial big-scale CNN model. It is considered the source of deep neural networks in computer visualization. That architecture got the picture Net Large-Scale visional Recognition Challenge (ILSVRC) in 2012. The difference amidst the AlexNet architecture and LeNet is incrementing the network layers, leading to a significantly more tunable parameter and the utilized regularization theory. It comprises five convolutional layers and three completely connected layers that classify an input picture into a single of the thousand classes of the ImageNet dataset and contain 1,000 categories. The overfitting issue is reduced by applying drop out after the first two connected layers, as shown in Figure 2.8. Besides, the use of ReLU after each convolutional and connected layer aims to improve the training efficiency matched to the traditionally utilized tanh function [19].

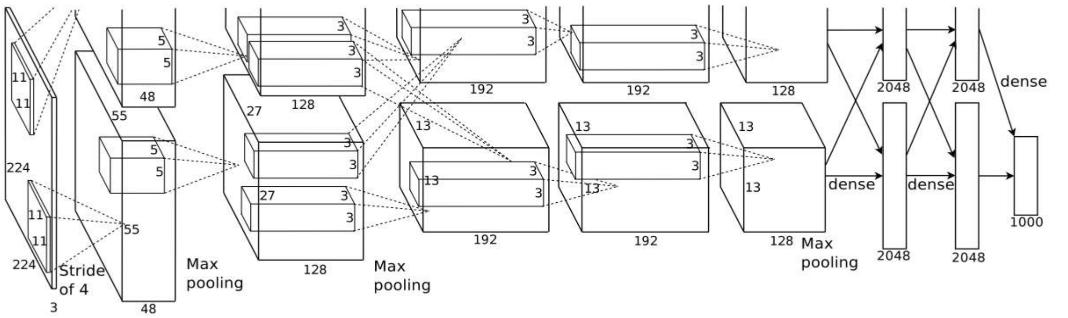

Figure 2.8 AlexNet Architecture

2.1.2.3 VGGNet

The VGGNet has been one of the comprehensive used CNN architectures since its introduction in 2014 because it used small-sized convolutional kernels that made the network deeper. The VGGNet uses two of 3x3 convolution kernels followed by a ReLU, max-pooling, and a set of three fully connected layers.

The use of smaller kernels reduces the number of factors. Therefore, it enables productive testing and training. Furthermore, by accumulating a group of 3x3 sized kernels, the active receptive area can be raised to more significant values (5x5 with two layers, 7x7 with three layers, etc.). With minimal filters, one can pile more layers causing profound networks, which leads to best achievement on vision mission. As in AlexNet, over-fitting is avoided by using activation dropouts in the primary two completely linked layers [20]. Figure 2.9 shows the best-performing model, VGGnet-16.

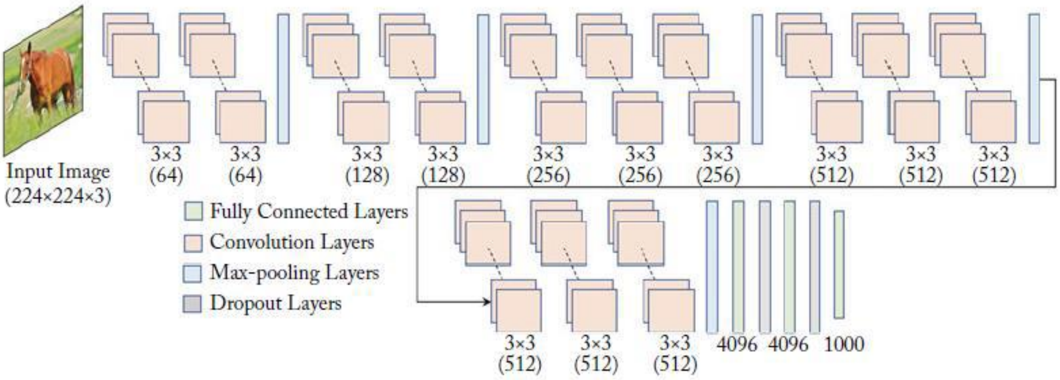

Figure 2.9 VGGNet Architecture

2.1.2.4 GoogleNet

GoogleNet is the first architecture that used a more complex architecture with many network branches in 2015. One of the best models in the ILSVRC'14 competition achieved an error rate of 6.7% on the classification task. However, GoogleNet is commonly known asthe "Inception Network" because the basic erection block of the network is arranged as a start module. In contrast to the sequential processing of previously discussed architectures, the processing in GoogleNet occurs in parallel. It means that all the fundamental processing blocks are placed in equivalent and combine their output feature representations, as shown in Figure 2.10. However, concatenating all the person's feature representations from every block along the deepness dimension causes a high-dimensional property output problem. The whole beginning module performs dimensionality reduction to avoid the feature output problembefore passagingthe input trait magnitude (say with remotenesss h * w * d) over the 3x3 and

5x5 convolution candidates. This remoteness lowering utilizes a fully joint layer equivalent to a 1x1 dimensional convolution operation. A giant network can be created by stacking multiple inception modules together. GoogleNet utilizes a typical global pooling pursued by a connected layer for classification toward the network's end. The global middle pooling layer maintains quicker computations with the best classification precision and a great-reduced numeral offactors [21].

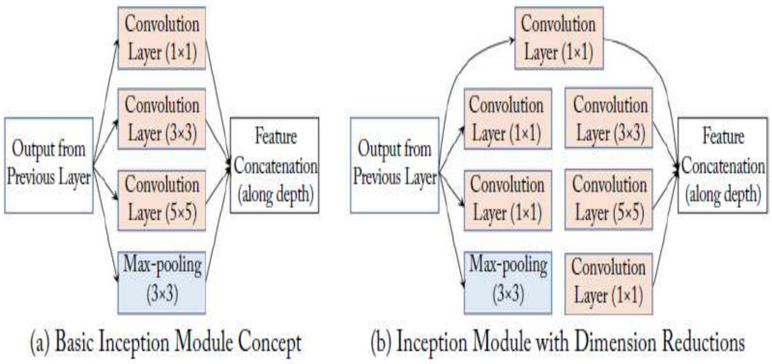

Figure 2.9 The Inception Module

2.1.2.5 ResNet

The Residual Network of Microsoft got the ILSVRC 2015 defy with a giant leap in performance by minimizing the error average to 3.6% compared with the last year's persuasive performance of 6.7% by GoogleNet. Furthermore, the skip identity connections in residual blocks are a remarkable feature in the residual network architecture that easily permits training deep CNN architectures, as shown in Figure 2.11.

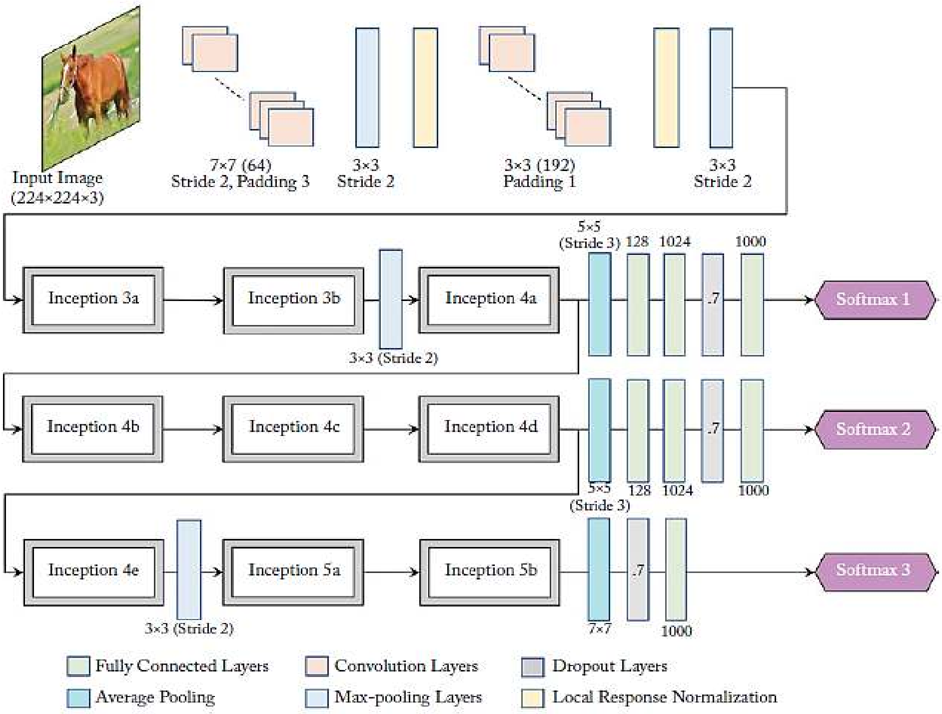

Figure 2.11 The Residual Network Architecture

In the remaining block, the new input is added to this diversion by directly linking the information, which avoids the diversion layers. This linking is named the surpass identity linking. In this way, the remaining block's conversion function is divided into an identity idiom (which characterizes the input) and a remaining idiom, which aid focus on the conversion of the remains advantage maps. In practice, architecture attains stabilized learning of actual profound models. The cause is that the residual feature mapping is predominately simpler than the unreferenced charting learned in traditional architectures [22].

2.2 Fall Detection Technologies Using Deep Learning

Deep learning algorithms have remained extensively utilized in diverses disciplines to solve challenges. Image classification [23], object detection, and scene recognition [24] tasks use convolutional neural networks. Human classification cases have the interest of deep learning researchers due to the outstanding performance of these algorithms. Deep learning algorithms have been applied to recognize actions in videos more recently. With the increased processing capabilities of GPUs and the giant video datasets made available to the research community, these applications are thriving.

The paper [25] suggested classifying falls from non-fall action in top observed kinect camera depth images. The use of depth camera photos is an excellent way to address privacy concerns. Furthermore, the top viewed camera output has the extra benefit of decreasing occlusion in a busy home. The approach used a constant backdrop setting and frame differencing throughout the study to acquire the foreground. The most related component selection was then used to extract the human silhouette. To extract feature vectors, they applied ellipse fit over a human silhouette. a bilateral support vector machine (SVM) classifier is employed to identify fallsfor non-falling frames. UR fall detection dataset was used to test the suggested approach. The whole fall recognition accuracy was computed as the normal of the 10-video classification accuracy and the entire testing accuracy of 96.3%.

[26] described a system for sensing human falls on furniture employing scene analysis according to deep learning and activity characteristics. The suggested method uses a deep learning algorithm named Faster R-CNN to analyze the scene to detect humans and furniture. Meanwhile, the human-furniture space relationship was discovered. The observed people's activity characteristics, such as human shape aspect ratio, centroid, and motion speed, are detected and tracked. Falls on furniture can be efficiently seen by evaluating changes in these qualities and judging the relationships between persons and tables nearby. The approach not only reliably and successfully recognized falls on furniture such as sofas and chairs but also discriminated them from other fall-like activities such as sitting or lying down, according to the findings of the experiments. The algorithm had a precision of 94.44%, a recall of 94.95 %, and an accuracy of 95.50 %.

A two-stage fall recognition system based on human posture features was proposed by [27]. They built the essential elements: aberration angles and spine proportion to represent the variations in human posture based on the humane skeleton retrieved by OpenPose as part of the preprocessing. The human body state was separated into three stages in the initial theatre, based on the mutable: tendency token and steady token combined by distributed key feature: stable state, fluctuation state, anddisordered condition. The ADL (activities of everyday living) behaviors with height settlement can be preliminarily rejected by determining whether the body is stable. They designed a time-continuous recognition technique in the second stage to further identify the disordered ADL actions and fall actions. When the humanoid body is constantly unstable, human posture features are used like contrast value, power value, and situation score are proposal to form a feature vector. Also, the classification algorithms such as support vector machine (SVM), K nearest neighbors (KNN), decision tree (DT), and random forest (RF) are used.

The results present that SVM of linear kernel function reached 97.34 % detection accuracy, 98.50 % precision, 97.33 % recall, and 97.91 % F1 score, respectively.

[28] suggested dividing the fall event into two parts: falling-state and fallen-state, which characterize the fall occurrences from dynamic and static perspectives. The object detection model (Yolo) and the human posture detection model (OpenPose) are utilized to acquire key points and human body position information. The dynamic properties of the human body (centroid speed, upper limb velocity) and static features of the human body are then extracted using a dual-channel sliding window model (external human ellipse). Then, to categorize the dynamic and static feature data independently, MLP (Multilayer Perceptron) and Random Forest are used. For fall detection, the categorization results are integrated. When tested utilizing the UR Fall observation Dataset and the Le2i Fall observation Dataset, the suggested approach obtains an accuracy of 97.33 % and 96.91 %, respectively.

[29] proposed a fall detection method based on a three-dimensional convolutional neural network (3D CNN) developed, which needs video kinematic data to train an automatic feature extractor and avoid needing a large fall dataset in a deep learning solution. The used 3D convolution could extract motion features from a temporal sequence, which is critical for fall detection since a 2D CNN could only encode spatial information. An LSTM (Long Short-Term Memory) based spatial visual attention method is used to help pinpoint the region of interest in each frame. The 3D CNN is trained using the Sports-1M dataset, which has no fall examples and then paired with LSTM to train a classifier with the fall dataset. The suggested technique achieves a 100% accuracy rate on the fall detection benchmark.

2.3 Related Method

There are five different types of fall detection systems. Radar-based methods, flexible device-based methods, and image processing-based methods are all examples of life information-based services.

2.3.1 Life Information-based Fall Detection Services

Many firms have developed devices that record users' daily activities to identify whether they are in danger. The data collected by these gadgets is sent to the user's relatives to help them determine whether the older person is living a normal life. The Zojirushi Corporation, for example, invented the"I-Pot," an intelligent electric kettle to identify the peculiar situation of celibate elderly people. Wi-Fi connects the kettle to the Internet, and information about the kettle's operating history is provided to relatives so they can see if the older person is using it. The older person may be at risk if they do not use the kettle for a long time. The Tokyo Gas Company offers a service that transmits the olderperson's daily usage history to the family to determine if the older person is in danger. The CICO Corporation gathers and analyzes the quantity of energy utilized in a home to assess whether or not the occupant is healthy [30]. These services are inexpensive and simple to use, but they do not provide any information about real-time fall detection.

2.3.2 Radio Wave-based Fall Detection Algorithms

There are two types of fall detection systems that use radio waves: traditional radio wave-based fall detection systems, such as radar-based fall detection systems, and fall detection systems based on Wi-Fi devices. These solutions do not infringe on user's privacy, and older people do not need to wear any equipment or sensors. The key advantages of a radar-based fall detection system are its high penetrating capabilities and resilient performance in adverse conditions. A dual-band radar-based fall detection system is shown in [31]. The most appropriate bands and the entire ground plane are optimized in this work to attain greater fall detection accuracy. This device can detect a fall accident within 4 meters when the radar is adjusted to 0 degrees. However, the system's fundamental flaw is that it can only see a few scenarios, such as falling and standing; many other postures in daily life are mistakenly identified as a fall accident [32]. To solve this problem, a radar-based fall detection system capable of detecting more poses than the systems is provided in [32].

Furthermore, this method can be used to categorize how older people fall, such as fainting, slipping, and so on. An Hidden Markov model is used to investigate the relationship between the moving rates of people and the radar frequency, and the experimental findings reveal that this method has a high detection rate. The approach of [33] introduces a novel radio locator wave-based fall detection system, which is made of two processing: pre-screen and classification. It is obvious that image processing-based methods and acoustical devices-based approaches. The first processing calculates the probability of a fall accident occurring, while the second processing segments the frequency information from wavelet characteristics at many scales. After categorization, a fall is detected. Their experiments revealed that the system developed in this work has excellent resilient performance and can see a restroom fall mishap. However, due to the determinations of machine learning, these methods are still limited in their detection accuracy. Thanks to deep understanding [34], they propose a fall detection system with greater accuracy. Many fall detection technologies based on the radar are employed in places where the radar signal is unrestricted, and there are no moving objects between the human and the radar, such as cats or dogs.

Furthermore, furniture relocation is also not taken into account [35]. Therefore, in terms of actual use, these systems may be limited. The fall detection systems based on Wi-Fi devices, called WiFall, were proposed [36]. The core idea behind this method is to identify falls using a sharp power-profile-decline pattern. The main benefit is that this system can detect falls using commonly used Wi-Fi routers without other equipment. Although Wi-Fi device-based algorithms are fantastic fall detection systems, there are still issues. They cannot see falls when the surroundings change or when other people are in the room.

2.3.3 Flexible Device-based Fall Detection Approaches

There are two kinds of fall detection systems based on flexible devices: wearable sensor-based systems and wearable camera-based systems are two types of wearable systems. The fundamental benefit of fall detection technologies based on flexible devices is that they may be used indoors and outdoors. However, many wearable sensor-based fall detection systems own been presented so far, owing to their low cost and lack of privacy concerns.The proposed fall detection system is based on a digital signal processor (DSP) and project backing equipment (PSE) [37]. The accelerometric signals that coincide to a time-based window of two next seconds and two preceding seconds to the effect immediate detected by the intelligent accelerometer sensor are analyzed by this system. In laboratory tests, this gadget demonstrated a high fall detection accuracy.

Moreover, this method allows for the extraction of some key features not found in another wearable sensor-based fall detection systems, as well as real-time processing. The algorithm described in [38] examines the accuracy of wearable sensors to determine the various causes of falls. A total of sixteen persons took part in the experiment, which involved falls caused by stumbles, slips, and other forms of imponderables. This algorithm takes acceleration data and classifies the causes of falls using linear discriminant analysis. This research outperformed both the two-sign combination method and the single sign method in terms of allergy in identifying the causes of falls. Approach [39] provided a new semi-automatic training paradigm extraction method for identifying fall detection and statussurveillance characteristics. This approach also included a two-layer network for surveillance non-fall activity and detecting fall events. Non-fall events and four categories of fall events were used in the experiments, and the results showed that this method effectively detects falls. The fall detection system suggested uses the backward wrapper technique to accomplish feature selection, which concerns accuracy and power efficiency. According to the findings of the experiments, this strategy saves 75.3 % of the power while maintaining a 92.9 % accuracy. However, because sensor-based fall detection necessitates constant analysis, it is difficult to predict how elderly people will fall.

Furthermore, these techniques can only be used in the event of an unexpected fall on a hard surface, to put it another way. Therefore, it is essential to conduct more research using data from actual falls [38]. Traditional fall detection approaches based on machine learning, and wearable sensors typically collect data using a non-overlapping or overlapping sliding window and then classify it.

Feature extraction and clustering were used to distinguish between fall and non-fall occurrences. However, some crucial information about the fall may be lost due to these algorithms; machine learning-based fall detection system activated by a fall accident [40]. The proposed method uses a buffer to store models as a preprocessing step before detecting multiple peaks to determine the trigger time. Then, fall/non-fall occurrences are categorized by comparing the models in the buffer. Another benefit of this method is that it is only 80% the cost of a non-overlapping skidding -window-based process and 78% the cost of an overlapping skidding window-based plan to compute. Conversely, the precision is as yet insufficient for feasible usage.

The approach of [41] combines a nonlinear filter with a Kalman filter to reduce fall-event detection misdetection. The process of [42] collects real data from older people and compares diverse methods to get better the fall detection rate. Deep learning and cloud computing are used in [43], and a data-augmentation technique is developed to boost its performance. [44] proposes an intelligent watch-based application that gathers real-world data from helpers to train a fall discovery model. This project also collects problems for practical application, such as diverse people's behaviors and battery life. Due to the scarcity of publically accessible data collections for sensor-based fall discovery, it provides an accelerometer and single gyroscope-based fall detection data collection, a well-constructed data set for sensor-based fall detection. This data collection includes 19 kinds of everyday activities and 15 fall activities. The participants in the other data sets are young people. However, in this data set, 14 actual senior people above the age of 60 are documented. References [45], [46], and [47] suggest wearable camera-based fall discovery approaches that involve attaching a camera to a waistline and detecting falls by evaluating changes in the images taken with the camera. This unique system compiles the benefits of vision-based and wearable sensor-based approaches. It is, nevertheless, lacking in expressions of particularity. Commercial wearable devices, like Apple Watch [48], Medical Guardian Fall Alarms [49], and Philips Lifeline Auto Alarms [50], have been created because wearable sensor-based systems may be used indoors or outdoors, and the sensor is low cost.

2.3.4 Pressure Sensor-based Fall Detection Methods

Although fall detection systems based on flexible devices operate effectively, the elderly may not be protected fromthe darkness. For example, when elderly people go to the bathroom at night, they may condone taking their electronics or wearable cameras. In this circumstance, the older person cannot be monitored in low-light settings at night, even though low-light conditions may be a significant cause of fall accidents. Fall detection systems based on pressure sensors are an ideal answer to this problem because older people do not need to wear any gadgets. The approach is given by using the NFI floor to track an older person and then calculating the attributes related to that senior person [51]. The method presented in this work uses features such as the longest size, the sum of magnitudes, and the duration of observation to evaluate the olderperson's postures and identify falls. The system presented uses compressing mate and negative infrared sensors to detect falls. These sensors exist in two states. State 0 is turned off, whereas State 1 is turned on. A simulator generates the data for ten older people, creating a data set. The states of squeeze mate sensors and passive infrared sensors are set to 0 when an aged person suffers a fall accident, and the fall accident is detected. Fiber-optic sensors are used in the approach to see the falls. These sensors collect pressure data, which is then used to generate a histogram that can be used to detect older people's activity. A fall accident is discovered when an aged person lies down for more than a certain amount of time. The algorithm combines compression sensors and accelerometers unobserved beneath intelligent tiles to reveal falls because squeeze sensor-based approaches cannot distinguish amidst falls and sleep on the floor. The experimental findings show that fall detection accuracy is enhanced. Fall detection systems based on pressure sensors are inexpensive and simple to use. On the other hand, pressure sensors are difficult to install and maintain. The solution to this challenge will be a fascinating future project.

2.3.5 Image Processing-based Fall Detection Systems

Fall detection has received much attention because of the low cost and non-intrusive nature of image processing-based. When elderly people go to the bathroom at night, they may forget to take their electronics or wearable cameras with them. As a result, even if image processing-based methods only work indoors, they are beneficial to the elderly. According to several studies, image processing-based fall detection approaches may be divided into color camera-based fall detection and depth camera-based fall detection. Fall detection systems that use a depth camera typically work better since the depth camera is practically unaffected by light. As a result, these devices function well in dim lighting and do not cause privacy concerns, and approaches based on depth cameras are gaining popularity. In comparison, many color camera algorithms can be utilized to successfully process images captured by a depth camera. As a result, this research examines image processing procedures by categorizing them into two groups based on how images are analyzed: skeletal analysis methods and shape analysis. Depth cameras are commonly used to record images for skeleton tracking systems because it is straightforward to extract the 3D joint coordinates of a person from depth photos. First, the ground plane equation and 3D coordinates of joints are calculated using the methods provided in [52] and [53]. Next, fall cases and non-fall cases are categorized based on their relationship. The support vector machine (SVM) is used in the fall detection system reported in [54] to assort the skeleton coordination of fall states and non-fall states, but merely walking, sitting, and downfall are considered. Next, it uses a method that calculates how the torso angle varies to characterize falls and fall-like events. Finally, it uses a technique that calculates the shifting of 3D joint coordinates to forecast the danger of falling. The precision of these methods is high; however, they difficultyrely on the skeleton detection outcome from Microsoft Kinet SDK or OpenNI, but they are not authoritative enough.

The study presents a method that detects the fall using the 3D position of the head and gets a high accuracy. However, the camera in this system is installed at such a low height, which may be blocked by furniture. Form analyzing is as well a perfect procedure for detecting falls. Bounding rectangle and approximately circumscribed ellipse-based fall detection were planned for more than ten years but are still efficacious in fall detection and have been improved. The fundamental disadvantage of these methods is that the camera orientation significantly impacts the bounding ellipse, and shape studies that use histograms as features have the same issue. In other words, if these methods are used, several cameras will be required to attain great precision. Fusibility of data acquired from various cameras is an alternate that has lately been researched to increase performance. They propose four cameras and logic algorithms to identify fall accidents in multiple orientations. In addition, numerous camera-based methods [55] are presented to detect falls in a wider room. Three-dimensional form analysis is also a viable option, but it comes at a considerable expense in computation.

Fall detection techniques can be split into two main groups; threshold-based technique or a machine learning-based technique. Threshold-based approaches detect it falls by inspection whether the measured feature exceeds a predefined threshold value or not. On the other hand, machine learning-based approaches use labeled data to train a classifier using supervised machine learning algorithms such as Support Vector

Machines (SVM), Decision Tree, and Artificial Neural Networks (ANN) to recognize the distinctive features of falls.

2.4 Classification of Fall Detection Technologies

For human fall detection systems to achieve their primary goal, they must be trained to distinguish between a human fall and the latest activities of quotidian living (ADL) (rambling, standup, sitting, and lying). Distinguishing between a fall and an ADL is difficult because many ADLs, such as lying on the floor or session from a standing position, are quite similar to fall. As a result, for these systems to perform correctly, they must be educated using falls and ADL data. Several sensors deployed in the surroundings, like as compression, floor-shaking, x-ray, microphones, and cameras, can collect this data. The data captured in acceleration measurements, compression signals, audio, or videosare then analyzed and provided to a classifier, determining if the data reflects a fall or an ADL.

Human fall-detection systems are divided into three groups based on the source of data collected [56], as shown in Figure 2.12; (1) wearable device-based system, (2) context-aware-based system (non-wearable device system), and (3) vision-based system. The performance of these three types of systems is sometimes described in idioms of accuracy (AC), specificity (SP) and sensitivity (SE), or precision (PR) and recall (RE) [57]. The model's capacity to properly recognize and assort an ADL as an ADL is defined as specificity. The model’s capacity to properly reveal and classify a fall as a fall is characterized by its sensitivity. The recall is another term for sensitivity. The capacity of a model to correctly recognize a fall and an ADL determines its accuracy. The number of accurately identified falls split by the number of falls detected yields precision.

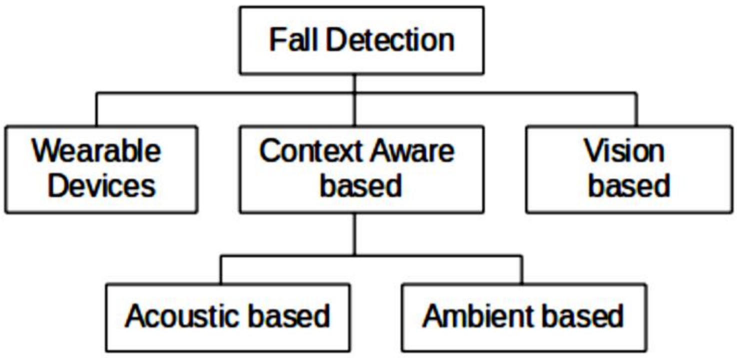

Figure 2.12 Classification of fall detection methods

2.4.1 Wearable Device-based Fall Detection Systems

Wearable fall detection devices is worn under or on the upper of a person's clothing [56]. These gadgets use a variety of sensors like as gyroscopes, accelerometers, tiltmeters, and oscilloscopes for detect a fall. The device must be kept on the individual at all times. As a result, the elderly gain location independence by walking around with the monitoring gadget [58]. However, there are situations when this is impossible, such as when the individual forgets to wearthe gadget, forgets to freightage the battery, or feels inconvenient wearing it. Because of these restrictions, researchers are focusing their efforts on developing contact-free fall detection systems that do not demand the user to wear anything extra to ensure their safety [59].

2.4.2 Context-aware Fall Detection Systems

These systems contain fall detection sensors placed in the surroundings of humans. Ambient and auditory sensors. These are examples of designs created using this technology like squeeze, floor, infrared, and mike. These systems acquire data using a combination of the sensors mentioned above. A dedicated PC is also connected to them. The data is transferred to a computer for additional processing and analysis [60]. A computer algorithm determines whether or not a fall has occurred based on particular threshold values and conditions.

According to machine learning classification algorithms, human activity is classified into fall and non-fall. Furthermore, systems advancedbeneath thisclass are split into two subcategories relyingon the kind of data raised via these sensors: (i) ambient-based fall detection systems aimed atstatistical sensor data (ii) acoustic-based fall detection systems for auditory data.

2.4.2.1 Ambient-based Fall Detection Systems

This subsection comprises systems that use environmental sensors such as a floor sensor, a pressure sensor, or an infrared sensor [61]. Region and pressure sensors must be put on the ground to read frequency of vibrations caused pending the fall. An X-ray sensor is used in conjunction with them to expose move in the circumferenceing environment. These systems are environment-reliant on because every house has a distinct flooring. As a result, particular configurations are necessary for its setup, making installation complicated.

2.4.2.2 Acoustic-Based Fall Detection Systems

Multiple microphones are deployed in the environment to collect sound emitted during a fall in this sort of system. Systems determine whether or not a fall has occurred by comparing the generated sound amplitude and direction [62] with a specified threshold value. Single of the most significant benefits of these systems is that no somatic touch is demand. Other characteristics comprise great precision and simple, low-cost composition. The most important disadvantage of these devices is such they bind people to the site where they are installed once they are established. Furthermore, they are inapplicable, not a regular living setting, due to the tremendous noise of completing any routine work. Finally, the feature of the indication received by the tracer is degraded by these sounds, which might lead to false fall detections.

CHAPTER III

MATERIALS AND METHODS

3.1 Introduction

This study proposes an algorithm that depends on a fall detection strategy to enhance independent living for seniors out of the analysis of the human movement. Visual-based monitoring systems can provide information on fall patterns. Falls can be detected utilizing intelligent monitoring systems and an automatic fall detection algorithm. The threshold-based method is extensively used to detect falls, but it necessitates a learning step to classify falls.

The methods that have been used in this study will be described in the same sequence as they have been applied: Dataset, Proposed Architecture, Data Normalization, Leaky Rectified Linear Unit, Sigmoid Function, and Cost Function**.**

3.2 Dataset

3.2.1 Dataset Resource

The UR Fall Detection dataset [63] embraces forehead and from above video sequences acquired by two Kinect cameras. The prime camera was seated at 1 m from the floor, while the second was 3m from the ceiling: two Kinect sensors and an accelerometer record 30 fall scenarios. The daily activities include: walking, sitting, bending, picking up an object from the floor, and crouching down. The simple activities (30 scenarios) were gathered using one Kinect sensor equivalent to the floor and ten sequences with fall-like activities such as fast lying on the floor and lying on the bed. Five participants were asked to implement daily activities and simulatedfalls.

The dataset was registered at 30 frames per second. Therefore, the number of images in the video sequences consisting of fall events is equal to 3000, while the number of pictures in video sequences with ADLs is equal to 10000.

Falls were mannered in various directions in connection with the camera view. Other fall accidents were recorded to contain forward, backward, and sideward falls. Participants outright fall from standing status and from sitting on the seat. Frame specimen taken from the UR Fall Detection dataset are shown in Figure 3.1.

Only video data from the RGB camera are used for the proposed fall detection approach. Table 3.1 represents the characteristics of the UR Fall detection dataset.

The composed video data set estimates the proposed fall detection accession by gathering daily activities and falls. In the present study, to value the detection execution of the proposed fall detection approach, experiments are conducted on the UR Fall Detection dataset .

First, the setup of the investigation for data collection is discussed. Then, various scenarios for daily activities and falls are described in this chapter. Finally, the collected data is used to train and test the proposed approach for fall detection.

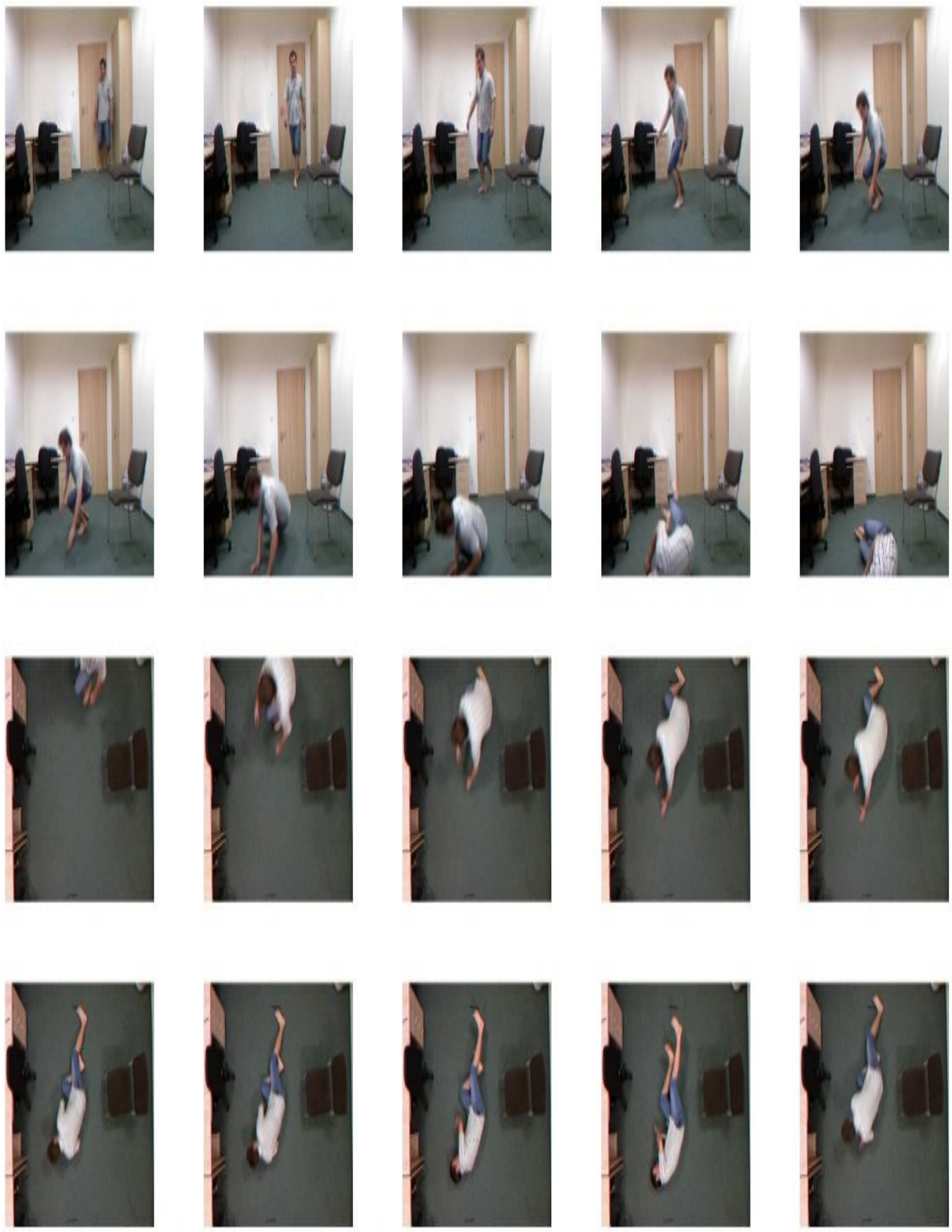

Figure 3.1 UR fall detection dataset.

Table 3.1 Features of UR fall detection.

Features	UR Fall detection dataset
Normal activities	Some normal activities contain walking, sitting, lying, curvaturing, and crouching down.
Falls	Several different types of falls.
Participants	Several participants with different clothing.
Lighting	Different light source represents variable illumination.
Setting	With only a small quantity of furniture, several different areas, including a living room and an office.
Camera perspective	Two perspectives.
Occlusions	A human is partially given the camera in a few video sequences.
Video length	Short video sequences (2 - 13 sec).

3.2.2 Dataset Preprocess

Preprocessing was achieved in two steps: scaling and augmentation. Images are scaled to 224 x 224 to utilize the Dense Net 121 model with top part layers and to reduce computational costs. Augmentation is applied to the scaled image, which modifies images before the training. We used brightness by a random factor is used to modify the brightness of images in order for our model to be trained to detect falls and not to fall in low-light rooms, horizontally flip an image at random (left to right), and clip (from 0 to 1) to enhance the image to expand the range of images of falls and non-falls that our model recognizes.

3.3 Proposed Architecture

3.3.1 Pretrained Deep Learning

It is a set of algorithms, pretrained models, and applications that form a framework enabling the users to design and implement deep neural networks CNNs and LSTM networks to perform classification, regression, and segmentation on picture, time-series, and text data. Deep learning toolbox applications and plots support the user to visualize network activations, edit network architectures, and watch training advance. The user can perform transfer learning with pretrained deep network models (including SqueezeNet, Inception-v3, ResNet-101, GoogLeNet, VGG-19, and DenseNet) models imported from TensorFlow™-Keras and Caffe.

3.3.2 Loading Pretrained DenseNet

DenseNet [64] as shown in Figure 3.2 is a network architecture in which each layer is feed-forward connected to each other (within each dense block). All previous layers' feature maps are handled as distinct inputs for each layer, while their own feature maps are transferred on to whole following levels as inputs. DenseNet 121 as shown in Figure 3.2 consists of 4 dense blocks. All blocks are a ReLU activation function except the last layer is the Softmax activation function. The Table 3.2 shows the details of DenseNet 121 architecture.

Table 3.2 DenseNet 121 architecture.

Layers	DenseNet 121
Convolution	7 x 7 conv, stride 2
Pooling	3 x 3 max pool, stride 2
Dense Block 1	$\begin{bmatrix} \text{1 x 1 Conv} \\ \text{3 x 3 Conve} \end{bmatrix}$x 6
Transition Layer 1	1 x 1 conv
	2 x 2 average pool, stride 2
Dense Block 2	$\begin{bmatrix} \text{1 x 1 Conv} \\ \text{3 x 3 Conve} \end{bmatrix}$x 12
Transition Layer 2	1 x 1 conv
	2 x 2 average pool, stride 2
Dense Block 3	$\begin{bmatrix} \text{1 x 1 Conv} \\ \text{3 x 3 Conve} \end{bmatrix}$x 24
Transition Layer 3	1 x 1 conv
	2 x 2 average pool, stride 2
Dense Block 4	$\begin{bmatrix} \text{1 x 1 Conv} \\ \text{3 x 3 Conve} \end{bmatrix}$x 16
Classification Layer	7 x 7 universal average pool
	1000D quite -connected, SoftMax

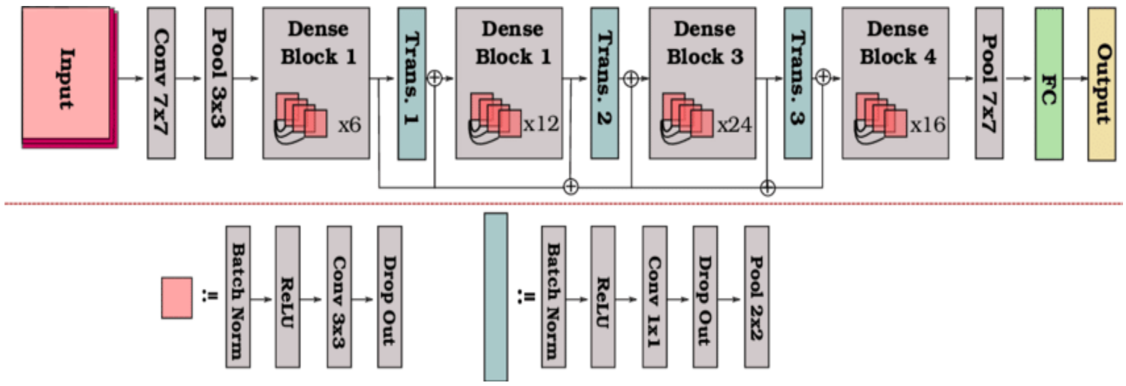

Figure 3.2 The Original DenseNet 121 Architecture

3.3.3 Modified DenseNetArchitecture

Shorter connections between layers close to the input and those close to the output allow convolutional networks to be significantly deeper, more accurate, and efficient to train [46].

It appreciated this insight and introduced the Dense Convolutional Network (DenseNet), a feed-forward network that connections each layer to every other layer. This network has L(L+1)/2 direct connections, whereas standard convolutional networks with L layers have L connections—one between each layer and its subsequent layer. All previous layers' feature-maps are utilized as inputs into each layer, and its own feature-maps are used as inputs into all subsequent layers. DenseNet have a number of compelling advantages, including the elimination of the vanishing-gradient problem, improved feature propagation, feature reuse, and a significant reduction in the number of parameters.

The modifications made to this architecture are by deleting the last layer (SoftMax layer) and adding some layers instead of it as shown in Table 3.3 and Figure 3.3. All layers are a ReLU activation function except pre the last layer is the Leaky ReLU activation function and the last layer is the Sigmoid activation function.

Table 3.3 Modified DenseNet 121 Architec.

Layers	DenseNet 121Modified
Pooling	medium pool
Dense	units: 256 activation function: leaky relu
Dropout	0.4
Dense	units:1 activation function: sigmoid

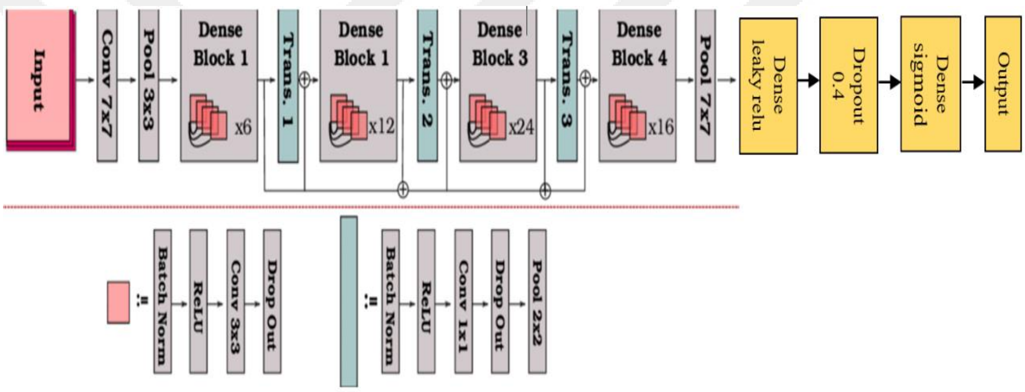

Figure 3.3 The Proposed Modified DenseNet Architecture

3.3.4 Pooling Layer

The pooling layer is used to make the input image smaller. A convolutional layer is typically followed by a pooling layer in a convolutional neural network. Pooling is used here to speed up the computation and identify more reliable features. For the pooling layers in our models, we selected the Average-pooling as shown in Figure 3.4. Average pooling is a technique for calculating the average and producing a simplified map.

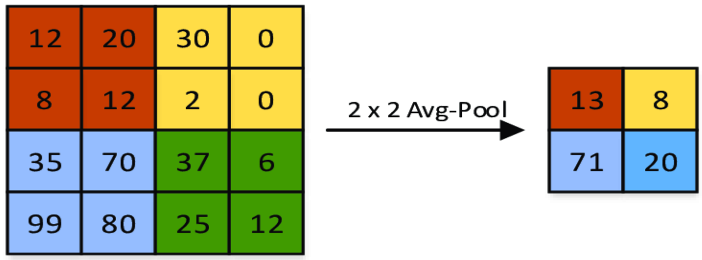

Figure 3.4 Average pooling

3.3.5 Dense Layer

A dense layer is one that is deeply connected to the layer before it, meaning that each of the layer's neurons is connected to every neuron in the layer before it. In neural networks, this layer is the most widely utilized layer. In a model, the dense layer's neuron receives input from every neuron in the preceding layer, and the dense layer's neurons conduct matrix-vector multiplication. The row vector of the output from the preceding layers is identical to the column vector of the dense layer in matrix vector multiplication. In matrix-vector multiplication, the row vector must have the same number of columns as the column vector.

3.3.6 Dropout

Dropout is a training strategy in which randomly selected neurons are rejected. They are "disappeared" at random. This means that on the forward pass, their contribution to the activation of downstream neurons is removed temporally, and on the backward pass, any weight updates are not applied to the neuron.

Neuron weights settle into their surroundings within a neural network as it learns. Neuronal weights are set for specific characteristics, allowing for some specialization. Neighboring neurons come to rely on this specialization, which can lead to a brittle model that is overly specialized to the training data if taken too far. Complex co-adaptations allude to a neuron's reliance on context during training. If neurons are taken out of the network at random during training, other neurons will have to step in and handle the representation needed to make predictions for the missing neurons. The network is thought to learn numerous separate internal representations as a result of this. As a result, the network becomes less sensitive to the weights of individual neurons and that the network will be able to generalize better and will be less prone to overfit the training data.

3.4 Data Normalization

It is the process of normalizing the input data to make it of the same scale across all dimensions as shown in Figure 3.5. Normalizing is achieved in the preprocessing phase. The most used technique to perform data normalization is to divide the data across each dimension by min-max feature scaling. However, the normalized data implicate them in the same scope as the activation function (usually 0 to 1). This reduces the frequency of non-zero gradients during training, resulting in faster learning of neurons in the network.

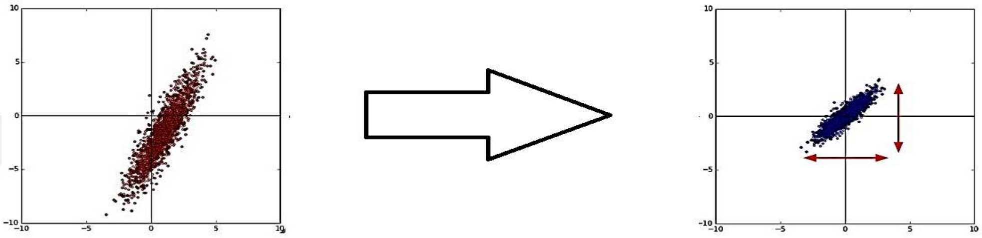

Figure 3.5 The Impact of the Normalization on the Input Data

3.5 Leaky Rectified Linear Unit

The Leaky Rectified Linear Unit, or Leaky ReLU, is a type of activation function dependent on the ReLU. Still, it has a slight slope instead of a flat slope for negative values, as illustrated in Figure 3.6. The slope coefficient is calculated before training rather than being learned during training. This form of activation function is commonly used in jobs with sparse gradients.

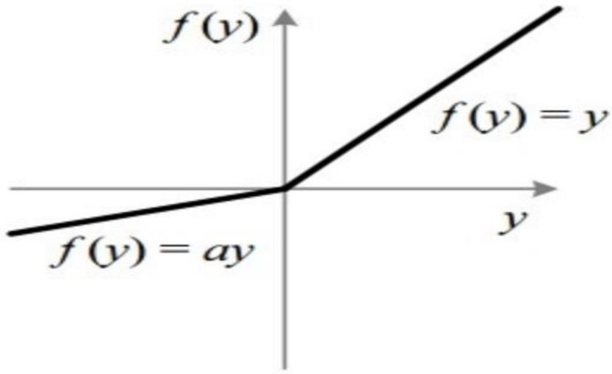

Figure 3.6 Leaky Rectified Linear Unit

3.6 Sigmoid Function

It's an arithmaic function as expressed in Equation 3.1 with the feature of transferring each real worth to a number among 0 and 1 in the letter "S" shape, as shown in Figure 3.7.

$$S = \frac{1}{1 + e^{-x}} \tag{3.1}$$

As an outcome, if the worth of S accesses positive infinity, the expected value of S becomes 1; if it accesses negative infinity, the foretelled value of S becomes 0. Suppose, Therefore, the sigmoid function's consequence is greater than 0.5. In that case, the label is categorized as positive category 1, and if it is less than 0.5, it is categorized as a negative class or labeled as category 0.

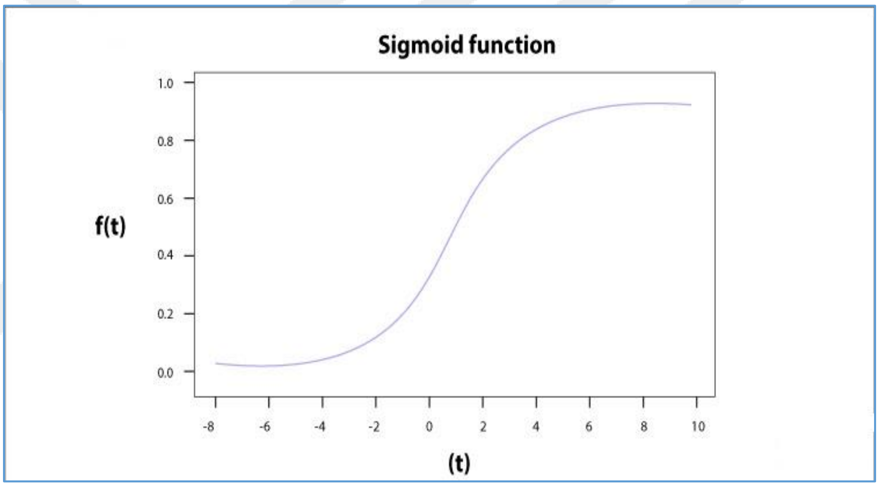

Figure 3.7 Sigmoid Function

3.7 Cost Function

The data knowledge expansion lifespan stage in which consultants try to match the best collection of weights and partiality to a machine-learning algorithm to reduce a loss function over the prediction domain is known as model training. Model training is to provide the best mathematical representation of the relationship amidst data features and aobjective label (in supervised learning) or between the elements themselves (in unsupervised learning) (unsupervised learning). Loss functions are essential for model training since they dictate how machine learning algorithms should be optimized depending on the goal, data type, and algorithm.

The cross-entropy function is the popularly used function for image classification as expressed in Equation 3.2. It is the same as mean square error, which deals with the distribution distinctions of two exemplifications.

$$D(S, L) = -\sum_i^c L_i \log S_i \quad (3.2)$$

where C refers to the digits of classes, S is the foretelling (output from the SoftMax), and L is the ground reality label. In the process classification, the cross-entropy for each pixel is computed and reduced to a scalar. one must look at equally class members where Li = 1 and non-class members Li = 0. Thus, the Equation becomes Equation 3.3.

$$D_{loss}(S, L) = -\sum_i^c L_i \log S_i - \sum_i^c (1 - L_i) \log(1 - S_i) \quad (3.3)$$

It's time to train the model after constructing, assembling, and setting the training options. But, first, we plot the loss curve of performance on the training and validation sets, as illustrated in Figure 3.8, to ensure that training has converged under such conditions.

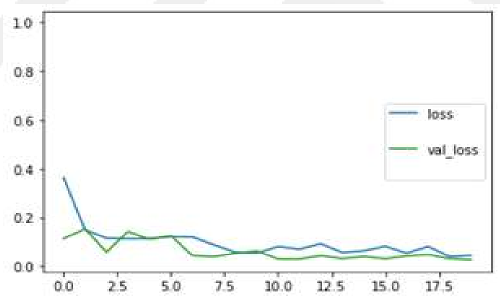

Figure 3.8 Training and validation loss function

3.8 Training and Testing the Network

The implementation of the testing and training was performed on the UR dataset using the machine of a configuration of AMD Ryzen 5 3600k Eight-core 3.5 GHz Processor, 32 GB RAM, NVIDIA GTX 1080 of 16 GB GPU memory. The model built here-in was built using GPU-enabled Python 3.6 and Keras 2.2.1 and demonstrates how the network architecture will affect the classification of images. We have also used Google Colab for faster execution. Google Colab was created to give anyone who needs GPUs to build a machine learning or deep learning model-free access to them. Google Colab can be thought of as a more advanced version of JupyterNotebook.

CHAPTER IV

RESULTS AND DISCUSSION

4.1 Introduction

This study presents innovative camera-based fall detection systems to help older individuals live independently. The proposed method extracts properties from RGB images that indicate the human body's motion to recognize falls.

The suggested design's accuracy, precision, recall, and F1-score are also determined. The next equation [65] displays the average of correctly matched data accuracy:

$$\text{Accuracy} = (TP + TN) / (TP + TN + FP + FN) \qquad (4.1)$$

where TP denotes a valid positive rate, implying that the detected output is a fall, in this case. TN refers to valid negative rates, which means the detected state is a non-fall, an actual non-fall event in real-time, while FP stands for false-positive rate, which means the seen status is a fall, an actual non-fall event in real-time. Finally, FN indicates a false negative, suggesting that the detected product is a non-fall.

In binary classification, a precision score of 100% implies that every unit in the positive group is linked to the positive unit, as specified on the basic formula [65]:

$$\text{Precision} = TP / (TP + FP) \qquad (4.2)$$

Sensitivity is known as the prospect of a positive check result using the rule [65]:

$$\text{Sensitivity} = TP / (TP + FN) \qquad (4.3)$$

The rule for counting specificity [65] is offered below, which donates the potential of a negative exam outcome.

$$\text{Specificity} = TN / (TN + FP) \qquad (4.4)$$

The F1-score performs the harmonic mean of precision and sensitivity. The F1-score of the test data is counted using the rule [65].

$$\text{F1 - score} = 2 * (\text{Precision} * \text{Sensitivity}) / (\text{Precision} + \text{Sensitivity}) \qquad (4.5)$$

4.2 Performance Evaluation

To assess how well the proposed model performs and to check the model’s capacity to recognize falls correctly, the given model's evaluation was done using the accuracy metric. The accuracy of a model is a metric that measures how well it works across all classes. When all of the different varieties are equally vital, it is beneficial. It is calculated using the ratio of correct estimates to total projections.

We performed various tests on the training and validation datasets to evolve the accuracy, and 98.83% accuracy is gained in the UR fall observation dataset. The model earns good accuracy for training and validation at 80% training, 10% validation, and 10% for testing data in the 20th epoch, as described in Figure 4.1 Preprocessing was completed in two steps: scaling and augmentation. Images are first resized to 224 by 224 pixels to reduce arithmetic costs. The adjusted image is then boosted, which alters the images before training. Finally, the idea is enhanced with brightness, horizontal flip, saturation, and clip;this makes the model's generalization simple.

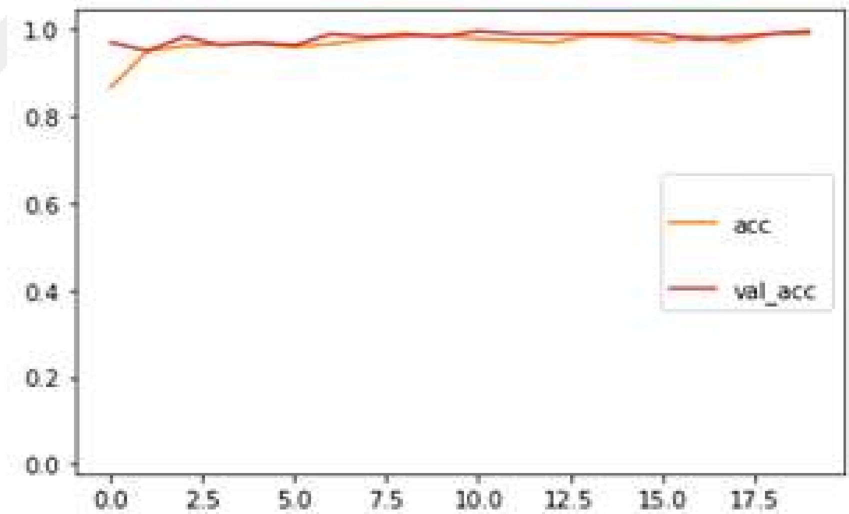

Figure 4.1 Training and validation evaluation

Our suggested model obtains perfect accuracy with fewer training epochs. The confusion matrix in Figure 4.2 demonstrates it.

The grey crates 84 and 86 point to the true positive and true negative, respectively. The proposed process fulfills a precise classification of 170 images out of 172 images. Between all, two non-fall images are classified as a fall case in the wrong way.

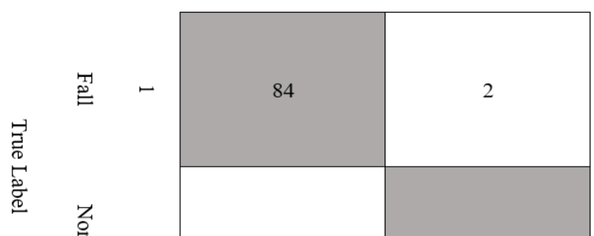

Figure 4.2 Confusion matrix output

4.3 Results

4.3.1 Comparison of The Proposed Model with Some Pretrained Models

Most of the pre-trained network topologies were compared to the suggested fall detection approach. Results utilizing only RGB photos on the UR Fall Detection dataset. Table 4.1 shows an overview of test accuracy for the various traditional systems, with VGG 16 and VGG 19 achieving 98% test accuracy and AlexNet achieving 98.3% accuracy albeit with a far larger number of parameters than our suggested model. LeNet, GoogleNet, and ResNet were 95%, 96%, and 97% accurate, respectively. Because VGG 16, VGG 19, and AlexNet are pre-trained models, their number of parameters is derived from [66], while the rest is determined experimentally as showed in Table 4.2.

Table 4.1 The proposed model's test accuracy compared to traditional models in the UR fall detection dataset for human fall classification.

Models	Accuracy %
VGG 16	98%
VGG 19	98%
LeNet	95%
AlexNet	98.3%
GoogleNet	96%
ResNet	97%
Proposed Method	98.83%

Table 4.2 Comparison of the number of parameters, depth, and training duration of existing human fall classification models.

Models	No of Parameters
VGG 16	138,357,544
VGG 19	143,667,240
AlexNet	60 million
GoogleNet	4 million
ResNet	60.3 million
Proposed Method	Total params: 7,300,161 Trainable params: 262,657 Non-trainable params: 7,037,504

4.3.2 Comparison of The Proposed Model with Pooling Layers

Pooling layers are fairly basic in the area of deep learning since typically frequently employ the maximum, minimum, or average values of input to down sample the data. Max polling aids in the extraction of low-level characteristics such as edges and points from data. To remove the most irrelevant characteristics from the data, min pooling is necessary. The layer chooses the average values of the elements accessible in the patch of the feature map when using average pooling. We get some translation invariance through average pooling. The smooth features may be extracted with the aid of average pooling. The average pooling layers appear as a composite of all colors present in the feature map's covered region. We tested different pooling functions and the results are summarized in Table 4.3. We chose average pooling in our suggested model.

Table 4.3 Different pooling layers results

Pooling	Accuracy
Max	96%
Min	96.6%
Average	98.83%

4.3.3 Comparison of The Proposed Model with Activation Functions

The activation function is used to produce or specify a specific output for a certain node based on the input. That is, the activation function will be applied to the summation results. We measured the performances of different activation functions and they are shown in Table 4.4. We chose the Leaky ReLU activation function in our model.

Table 4.4 The Accuracy of correctly classified with activation functions

Activation Functions	Accuracy
Tanh	87.65%
ReLU	91.45%
Leaky ReLU	98.83%

4.3.4 Comparison of the Proposed Model with Dropout

In deep layers, a dropout rate of 40% is employed because a dropout rate of less than 40% results in inferior accuracy as well as overfitting difficulties with training data. Figure 4.3 depicts the effect of the dropout rate on the change in accuracy. For the UR fall detection dataset, the suggested model achieves a maximum accuracy of 98.83 percent at a dropout rate of 40%, as shown in Figure 4.3.

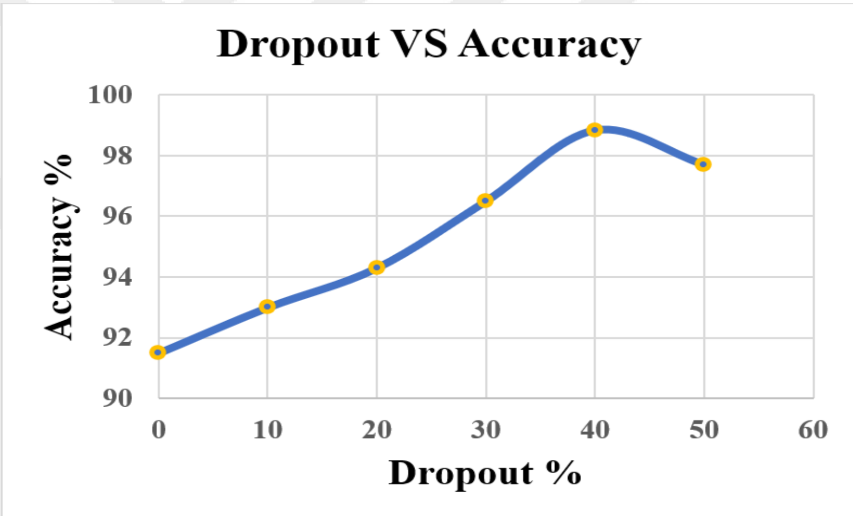

Figure 4.3 Dropout vs. accuracy curve

4.3.5 Comparison with the State-of-the-Art Methods

The proposed fall detection approach was compared with some other fall detection approaches with the same UR Fall Detection dataset using only RGB images.

Table 4.5 shows the comparative classification results of our proposed technique and five other deep learning methods on the UR Fall Detection Dataset. The data reveal that our method outperforms other methods even in terms of overall accuracy, precision, and F-score. This means that, when it comes to fall detection, our system can more accurately detect the probability of a fall while reducing the chance of mistaking other everyday activities for fall out.

Table 4.5 Results between our method and some approaches on the same dataset.

Method	Accuracy %	Precision %	F-Score
Dai et al. [67]	97.33	97.78	97.78
Harrou et al. [68]	96.66	94	96.91
Kwolek and Kepski [69]	90	83.30	90.89
Kasturi et al. [70]	96.34	-----	-----
Wang et al. [71]	96.91	97.65	97.08
Our method	**98.83**	**97.72**	**98.85**

4.4 Real Time Application

A video camera is installed in the residence. It also supplies us with information about our surroundings in a fall to keep an eye on the elderly person. Sequential frames are created using video with 224 x 224 pixels resolution. The process is illustrated in Figure 4.4. Finally, to evaluate the category, the output is submitted to a sigmoid classifier.

Figure 4.4 Real time application of fall detection system

4.5 Discussion

The work offered in this study has fundamentally concentrated on evaluating the proposed fall detection process based on computer vision techniques. The approximation presented here utilizes enhanced features that are extractored from the images. Then the features are fed into machine learning algorithms to classify falls.

The modified DenseNet completes superior classification execution in precision compared to the other strategies. In expansion, the modified DenseNet gives way better exactness and specificity, which response to its capacity to classify day-by-day exercises accurately and drop occasions.

CHAPTER V

CONCLUSIONAND RECOMMENDATIONS

5.1 Introduction

A fall detection system, which can grasp the situation of the elderly, detect the fall accident in actual time automatically and notify family members, hospitals, or health centers, would be essential for celibate elderly persons and play a vital role in the health care sector for elderly persons. This study presented a series of algorithms to solve the fall detection problems.

The presented work is a novel attempt to automatically inspect the fall incidents to give independent living for older adults in indoor environments.

According to the results obtained from the used technique, it can be found that distinguishing a fall action be contingent basically on the quality of the class inputs. Therefore, the lineaments of the extracted human have a vital turn in the effectiveness and sturdiness of detecting human falls. The suggested approach was based on visual elements to decide if a video sequence caused a fall.

In this thesis, we suggested modifying Dense architecture to detect human falls. This research aimed to investigate appropriate methods for detecting falls by analyzing the motion and shape of the human body. A video-based system for sensing falls is suggested to achieve the project aim. The proposed system consists of three steps: detecting moving objects in a frame, tracking such things from frame to frame, and analyzing object tracks to realize the behavior.

5.2 Recommendations

Further developments on the algorithm could be one of the future jobs of this research.

1. The proposed fall detection system is implemented to monitor one person at home. The model can be trained to monitor many people.

2. Multiple cameras in a single scene could be another idea for future work to increase the robustness of the proposed fall detection method. Although the obtained results for fall inspection are inspiring using only a single camera, the arrangement performance could be developed further by utilizing multiple cameras since a single camera limits the vision angle of the scene.
3. Instead of utilizing only the video data from information parts, some extra sensors like accelerometers could be extracted to raise the detection rate. Many sensors have been shown to give a better result when merged with the camera sensor. These kinds of devices will be considered for future projects. Again, though, this work was previously a step toward high-performance fall detection structures.

REFERENCES

[1] Mubashir, M., Shao, L., Seed, L. (2013). A survey on fall detection: Principles and approaches. *Neurocomputing,* **100**, 144-152.

[2] Yang, L., Ren, Y., Hu, H., Tian, B. (2015). New fast fall detection method based on spatio-temporal context tracking of head by using depth images. Sensors, **15(9),** 23004-23019.

[3] Rougier, C., Meunier, J., St-Arnaud, A., & Rousseau, J. (2011). Robust video surveillance for fall detection based on human shape deformation. *IEEE Transactions on circuits and systems for video Technology*, ***21*(5),** 611-622.

[4] Lord, S. R., Smith, S. T., Menant, J. C. (2010). Vision and falls in older people: risk factors and intervention strategies. *Clinics in geriatric medicine*, ***26*(4),** 569-581.

[5] Lin, C. W., Ling, Z. H. (2007, August). Automatic fall incident detection in compressed video for intelligent homecare. In *2007 16th International Conference on Computer Communications and Networks* (pp. 1172-1177). IEEE.

[6] Wang, X., Liu, H., Liu, M. (2016, December). A novel multi-cue integration system for efficient human fall detection. In *2016 IEEE International Conference on Robotics and Biomimetics (ROBIO)* (pp. 1319-1324). IEEE.

[7] Stel, V. S., Smit, J. H., Pluijm, S. M., & Lips, P. (2004). Consequences of falling in older men and women and risk factors for health service use and functional decline. *Age and ageing*, ***33*(1)**, 58-65.

[8] Fleming, J., &Brayne, C. (2008). Inability to get up after falling, subsequent time on floor, and summoning help: prospective cohort study in people over 90. *Bmj*, *337*.

[9] Zhang, B., Horváth, I., Molenbroek, J. F., &Snijders, C. (2010). Using artificial neural networks for human body posture

prediction. *International Journal of Industrial Ergonomics*, *40*(4), 414-424.

[10] Chaccour, K., Darazi, R., El Hassani, A. H., Andres, E. (2016). From fall detection to fall prevention: A generic classification of fall-related systems. *IEEE Sensors Journal*, ***17*****(3),** 812-822.

[11] Nielsen, M. A. (2015). *Neural networks and deep learning* (Vol. 25). San Francisco, CA: Determination press.

[12] Liu, W., Wang, Z., Liu, X., Zeng, N., Liu, Y., &Alsaadi, F. E. (2017). A survey of deep neural network architectures and their applications. *Neurocomputing***,** ***234***, 11-26.

[13] Schmidhuber, J. (2015). Deep learning in neural networks: An overview. *Neural networks*, *61*, 85-117.

[14] Xu, J., Wang, B., Li, J., Hu, C., Pan, J. (2017, June). Deep learning application based on embedded GPU. In *2017 First International Conference on Electronics Instrumentation & Information Systems (EIIS)* (pp. 1-4). IEEE.

[15] James, A. P. (Ed.). (2019). *Deep Learning Classifiers with Memristive Networks: Theory and Applications* (Vol. 14). Springer.

[16] Wu, L. (2019). *Biomedical Image Segmentation and Object Detection Using Deep Convolutional Neural Networks* (Doctoral dissertation, Purdue University Graduate School).

[17] Rawat, W., Wang, Z. (2017). Deep convolutional neural networks for image classification: A comprehensive review. *Neural computation*, ***29*****(9)**, 2352-2449.

[18] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, “Backpropagation Applied to Handwritten Zip Code Recognition,” vol. 1, no. 4, pp. 541-551, 1989.

[19] A. Krizhevsky, I. Sutskever, and G. Hinton, “ImageNet Classification with Deep Convolutional Neural Networks,”*Neural Information Processing Systems,* vol. 25, 01/01, 2012.

[20] Simonyan, K., Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.

[21] Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., ... &Rabinovich, A. (2015). Going deeper with convolutions. In *Proceedings*

of the IEEE conference on computer vision and pattern recognition (pp. 1-9).

[22] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770-778).

[23] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770-778).

[24] Zhou, B., Lapedriza, A., Xiao, J., Torralba, A., & Oliva, A. (2014). Learning deep features for scene recognition using places database.

[25] Kasturi, S., & Jo, K. H. (2017, July). Classification of human fall in top Viewed kinect depth images using binary support vector machine. In *2017 10th International Conference on Human System Interactions (HSI)* (pp. 144-147). IEEE.

[26] Min, W., Cui, H., Rao, H., Li, Z., & Yao, L. (2018). Detection of human falls on furniture using scene analysis based on deep learning and activity characteristics. *IEEE Access*, *6*, 9324-9335.

[27] Han, K., Yang, Q., & Huang, Z. (2020). A two-stage fall recognition algorithm based on human posture features. *Sensors*, *20*(23), 6966.

[28] Wang, B. H., Yu, J., Wang, K., Bao, X. Y., & Mao, K. M. (2020). Fall detection based on dual-channel feature integration. *IEEE Access*, *8*, 103443-103453.

[29] Lu, N., Wu, Y., Feng, L., & Song, J. (2018). Deep learning for fall detection: Three-dimensional CNN combined with LSTM on video kinematic data. *IEEE journal of biomedical and health informatics*, *23*(1), 314-323.

[30] Xiangbo, K. O. N. G. (2020). A Fall Detection System for ElderlyPersons.

[31] Soh, P. J., Mercuri, M., Pandey, G., Vandenbosch, G. A., Schreurs, D.M. P. (2012). Dual-band planar bowtie monopole for a fall-detection radar and Telemetry system. *IEEE Antennas and Wireless Propagation Letters*, 11, 1698-1701.

[32] Amin, M. G., Zhang, Y. D., Ahmad, F., & Ho, K. D. (2016). Radar signal processing for elderly fall detection: The future for in-home monitoring. *IEEE Signal Processing Magazine*, **33(2)**, 71-80.

[33] Su, B. Y., Ho, K. C., Rantz, M. J., Skubic, M. (2014). Doppler radar fall activity detection using the wavelet transform. *IEEE Transactions on Biomedical Engineering*, 62(3), 865-875.

[34] Jokanović, B., Amin, M. (2017). Fall detection using deep learning in range-Doppler radars. IEEE Transactions on Aerospace and Electronic Systems, 54(1), 180-189.

[35] Garripoli, C., Mercuri, M., Karsmakers, P., Soh, P. J., Crupi, G., Vandenbosch, G. A., ... Schreurs, D. (2014). Embedded DSP-based Telehealth radar system for remote in-door fall detection. *IEEE journal of biomedical and health informatics*, **19(1),** 92-101.

[36] Wang, Y., Wu, K., Ni, L. M. (2016). Wifall: Device-free fall detection by wireless networks. *IEEE Transactions on Mobile Computing*, **16(2**), 581-594.

[37] Estudillo-Valderrama, M. A., Roa, L. M., Reina-Tosina, J., Naranjo-Hernandez, D. (2009). Design and implementation of a distributed fall detection system—personal server. *IEEE Transactions on Information Technology in Biomedicine*, **13(6**), 874-881.

[38] Aziz, O., Robinovitch, S. N. (2011). An analysis of the accuracy of wearable sensors for classifying the causes of falls in humans. *IEEE transactions on neural systems and rehabilitation engineering*, **19(6)**, 670-676.

[39] Abeyruwan, S. W., Sarkar, D., Sikder, F., Visser, U. (2016). Semi-automatic extraction of training examples from sensor readings for fall detection and posture monitoring. *IEEE Sensors Journal*, **16(13**), 5406-5415.

[40] Putra, I. P. E. S., Brusey, J., Gaura, E., Vesilo, R. (2018). An event-triggered machine learning approach for accelerometer-based fall detection. Sensors, 18(1), 20.

[41] Sucerquia, A., López, J. D., Vargas-Bonilla, J. F. (2018). Real- life/real-time elderly fall detection with a triaxial accelerometer Sensors,18(4), 1101.

[42] Khojasteh, S. B., Villar, J. R., Chira, C., González, V. M., De la Cal, E. (2018). Improving fall detection using an on-wrist wearable accelerometer. Sensors**, 18(5**), 1350.

[43] Santos, G. L., Endo, P. T., Monteiro, K. H. D. C., Rocha, E. D. S., Silva, I., & Lynn, T. (2019). Accelerometer-based human fall detection using Convolutional Neural networks. Sensors, 19(7), 1644.

[44] Mauldin, T. R., Canby, M. E., Metsis, V., Ngu, A. H., & Rivera, C. C. (2018). SmartFall: A smartwatch-based fall detection system using deep learning. *Sensors*, ***18*(10)**, 3363.

[45] Ozcan, K., Mahabalagiri, A. K., Casares, M., Velipasalar, S. (2013). Automatic fall detection and activity classification by a wearable embedded smart camera. *IEEE journal on emerging and selected topics in circuits and systems*, ***3*(2),** 125-136.

[46] Ozcan, K., Velipasalar, S. (2015). Wearable camera-and accelerometer-based fall detection on portable devices. *IEEE Embedded Systems Letters*, ***8*(1),** 6-9.

[47] Ozcan, K., Velipasalar, S., Varshney, P. K. (2016). Autonomous fall detection with wearable cameras by using relative entropy distance measure. *IEEE Transactions on Human-Machine Systems*, ***47*(1),** 31-39.

[48] Burton, J. Healthcare Innovation: The Use of Fall Detectors to Detect Unwitnessed Falls and Reduce A Long Lie.

[49] Kong, X., Chen, L., Wang, Z., Chen, Y., Meng, L., Tomiyama, H. (2019). Robust self-adaptation fall-detection system based on camera height. *Sensors*, ***19*(17)**, 3768.

[50] Xiangbo, K. O. N. G. (2020). A Fall Detection System for Elderly Persons.

[51] Rimminen, H., Lindström, J., Sepponen, R. (2017). Positioning accuracy and multi-target separation with a human tracking system using near field imaging. *International Journal on Smart Sensing and Intelligent Systems*, *2*(1).

[52] Planinc, R., &Kampel, M. (2012, November). Robust fall detection by combining 3D data and fuzzy logic. In *Asian Conference on Computer Vision* (pp. 121-132). Springer, Berlin, Heidelberg.

[53] Planinc, R., Kampel, M., Ortlieb, S., Carbon, C. C. (2013). User-centered design and evaluation of an ambient event detector based on a balanced scorecard approach. *International Journal on Advances in Life Sciences Volume 5, Number 3 & 4, 2013.*

[54] Alazrai, R., Momani, M., Daoud, M. I. (2017). Fall detection for elderly from partially observed depth-map video sequences based on view-invariant human activity representation. *Applied Sciences*, **7(4),** 316.

[55] Rougier, C., Meunier, J., St-Arnaud, A., & Rousseau, J. (2007, May). Fall detection from human shape and motion history using video surveillance. In *21st International Conference on Advanced Information Networking and Applications Workshops (AINAW'07)* (Vol. 2, pp. 875-880). IEEE.

[56] Nizam, Y., Mohd, M. N. H., Jamil, M. M. A. (2016). A study on human fall detection systems: Daily activity classification and sensing techniques. *International Journal of Integrated Engineering*, *8*(1).

[57] Abbate, S., Avvenuti, M., Corsini, P., Light, J., & Vecchio, A. (2010). Monitoring of human movements for fall detection and activities recognition in elderly care using wireless sensor network: a survey. Wireless Sensor Networks: Application-Centric Design, 147-166.

[58] Dang, T. T., Truong, H., Dang, T. K. (2016). Automatic fall detection using smartphone acceleration sensor. *International Journal of Advanced Computer Science and Applications*, **7(12**), 123-129.

[59] Uddin, M., Khaksar, W., &Torresen, J. (2018). Ambient sensors for elderly care and independent living: a survey. *Sensors*, *18*(7), 2027.

[60] Popescu, M., Li, Y., Skubic, M., Rantz, M. (2008, August). An acoustic fall detector system that uses sound height information to reduce the false alarm rate. In *2008 30th Annual International Conference of the IEEE Engineering in Medicine and Biology Society* (pp. 4628-4631). IEEE.

[61] Toreyin, B. U., Soyer, A. B., Onaran, I., Cetin, E. E. (2007). Falling person detection using multi-sensor signal processing. *EURASIP Journal on Advances in Signal Processing*, *2008*, 1-7.

[62] Shojaei-Hashemi, A., Nasiopoulos, P., Little, J. J., Pourazad, M. T. (2018, May). Video-based human fall detection in smart homes using deep learning. In *2018 IEEE International Symposium on Circuits and Systems (ISCAS)* (pp. 1-5). IEEE.

[63] Kwolek, B., Kepski, M. (2014). Human fall detection on embedded platform using depth maps and wireless accelerometer. *Computer methods and programs in biomedicine*, *117*(3), 489-501.

[64] Huang, G., Liu, Z., Van Der Maaten, L., Weinberger, K. Q. (2017). Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 4700-4708).

[65] Ghoneim(2021), S. Accuracy, Recall, Precision, F-Score & Specificity, Which to Optimize on Available online:https://towardsdatascience.com/accuracy-recall-precision-f-score-specificity-which-to-optimize-on867d3f11124.

[66] Keras Applications. Available online: http://keras.io/api/applications(accessed on 20 March 2022).

[67] Dai, B., Yang, D., Ai, L., Zhang, P. (2018, October). A novel video-surveillance-based algorithm of fall detection. In *2018 11th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI)* (pp. 1-6). IEEE.

[68] Harrou, F., Zerrouki, N., Sun, Y., &Houacine, A. (2019). An integrated vision-based approach for efficient human fall detection in a home environment. *IEEE Access*, *7*, 114966-114974.

[69] Kwolek, B., &Kepski, M. (2014). Human fall detection on embedded platform using depth maps and wireless accelerometer. *Computer methods and programs in biomedicine*, *117*(3), 489-501.

[70] Kasturi, S., Jo, K. H. (2017, July). Classification of human fall in top Viewed kinect depth images using binary support vector machine. In *2017 10th International Conference on Human System Interactions (HSI)* (pp. 144-147). IEEE.

[71] Wang, B. H., Yu, J., Wang, K., Bao, X. Y., Mao, K. M. (2020). Fall detection based on dual-channel feature integration. *IEEE Access*, *8*, 103443-103453.

CURRICULUM VITAE

Name : Mustafa Hussein RAFEEQ

Education

- **M.sc. ELECTRICAL & ELECTRONICS ENGINEERING,**

  Graduated 2022
  University of Gaziantep / College of Engineering / Electrical & Electronics Engineering Dept. / Gaziantep/Turkey.

- **B.s.c Computer Engineering Techniques , Graduated 2015**

  AL-Qalam University /College of Engineering/ Computer Engineering Techniques Dept. / Kirkuk / Iraq.

- **Secondary school education, Graduated 1999**
  AL-Tamim High School for boys/ Science Division /Kirkuk /Iraq.

Publication:

- Mustafa Hussein RAFEEQ, Serkan ÖZBAY, "A Real-Time Fall Detection Of Elderly People In Indoor Environments", 3rd International Sciences and Innovation Conference, Cangress Book, November 2021.