IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 22 March 2023, accepted 5 April 2023, date of publication 11 April 2023, date of current version 27 April 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3266296

RESEARCH ARTICLE

Automated Image Annotation With Novel Features Based on Deep ResNet50-SLT

MYASAR MUNDHER ADNAN[1,2], MOHD SHAFRY MOHD RAHIM[3], AMJAD REHMAN KHAN[4], (Senior Member, IEEE), AHMED ALKHAYYAT[2], FATEN S. ALAMRI[5], TANZILA SABA[4], (Senior Member, IEEE), AND SAEED ALI BAHAJ[6]

[1]Faculty of Engineering, School of Computing, Universiti Teknologi Malaysia, Skudai, Johor 81310, Malaysia
[2]Department of Computer Technical Engineering, Islamic University, Najaf 192122, Iraq
[3]School of Computing, Universiti Teknologi Malaysia, Skudai, Johor 81310, Malaysia
[4]Artificial Intelligence and Data Analytics Laboratory, CCIS, Prince Sultan University, Riyadh 11586, Saudi Arabia
[5]Department of Mathematical Sciences, College of Science, Princess Nourah Bint Abdulrahman University, Riyadh 11671, Saudi Arabia
[6]MIS Department, College of Business Administration, Prince Sattam Bin Abdulaziz University, Alkharj 11942, Saudi Arabia

Corresponding author: Saeed Ali Bahaj (saeedalibahaj@gmail.com)

This research was funded by Princess Nourah bint Abdulrahman University and Researchers Supporting Project number (PNURSP2023R346), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia.

ABSTRACT Due to their vast size, the growing number of digital images found in personal archives and on websites has become unmanageable, making it challenging to retrieve images from these large databases accurately. While these collections are popular due to their convenience, they often need to be equipped with proper indexing information, making it difficult for users to find what they need. One of the most significant challenges in computer vision and multimedia is image annotation, which involves labeling images with descriptive keywords. However, computers need to possess the capability to understand the essence of images in the same way that humans do, and people can only identify images based on their visual attributes rather than their deeper semantic meaning. Therefore, image annotation requires keywords to effectively communicate the contents of an image to a computer system. However, raw pixels in an image need to provide more information to generate semantic concepts, making image annotation a complex task. Unlike text annotation, where the dictionary linking words to semantics is well established, image annotation lacks a clear definition of ‘‘words’’ or ‘‘sentences’’ that can be associated with the meaning of the image, known as the semantic gap. To address this challenge, this study aimed to characterize image content meaningfully to make information retrieval easier. An improved automatic image annotation (AIA) system was proposed to bridge the semantic gap between low-level computer features and human interpretation of images by assigning one or multiple labels to images. The proposed AIA system can convert raw image pixels into semantic-level concepts, providing a clearer representation of the image content. The study combined the ResNet50 and slantlet transform with word2vec and principal component analysis with t-distributed stochastic neighbor embedding to balance precision and recall. This allowed the researchers to determine the optimal model for the proposed ResNet50-SLT AIA framework. A Word2vec model with ResNet50-SLT was used with principal component analysis and t-distributed stochastic neighbor embedding to improve IA prediction accuracy. The distributed representation approach involved encoding and storing information about image features. The proposed AIA system utilized seq2seq to generate sentences depending on feature vectors. The system was implemented on the most popular datasets (Flickr8k, Corel-5k, ESP-Game). The results showed that the newly developed AIA scheme overcame the computational time complexity associated with most existing image annotation models during the training phase for large datasets. The performance evaluation of the AIA scheme showed its excellent flexibility of annotation, improved accuracy, and reduced computational costs, thus outperforming the existing state-of-the-art methods. In conclusion, this AIA framework can provide immense benefits in accurately selecting and extracting image features

The associate editor coordinating the review of this manuscript and approving it for publication was Mehul S. Raval.

and easily retrieving images from large databases. The extracted features can effectively be used to represent the image, thus accelerating the annotation process and minimizing the computational complexity.

INDEX TERMS Automatic image annotation, deep learning, features extraction, digital learning, Slantlet transform, technological development.

I. INTRODUCTION

Confucius' quote highlights the importance of images in our lives "A picture is worth a thousand words". Digital images have become a ubiquitous presence in both professional and personal lives. They are used in medical, insurance, advertising, commerce, and personal events such as birthdays and trips. This widespread use of digital images has resulted in an exponential increase in their number, with billions of images being stored on specialized websites. Searching for an image from a large database can be challenging, leading to the development of various methods for rapid and precise image retrieval. Besides basic visual features like color and texture, semantic labels can also be utilized. While low-level visual functions allow for fast retrieval, using a query image as input may only sometimes be practical for users The search of an image from a huge database is undoubtedly a very complex task. To overcome such problems, numerous methods have been developed for accessing the right image rapidly and precisely [1], [2]. Retrieving a digital image is shown through the use of either its low-level visual elements like shape, color, and texture or its semantic labels or keywords. A user can search for similar images by utilizing low-level visual features and receive a collection of visually similar images by presenting a reference image. Although users can often locate the desired image through this method, it is not always a guaranteed outcome.

The significant contributions of this article are summarized below.

1-Generated new features vectors involving ResNet50-Slantlet transform, which increases the accuracy while maintaining a higher level of image retrieval.

2- Enhance the performance of image coding and annotating of the proposed AIA scheme by designing a decomposition method while maintaining prediction image in AIA.

3-Developed a new AIA system with clean descriptions and semantic relationships between vectors for image retrieval and description.

The following sections of the manuscript provide. Section II provides background information on previous research. In Section III, an advanced deep feature extraction approach is described. The article's main focus is in Section IV, where a new method for image annotation is proposed. This new method is evaluated against other techniques such as MBRM [3], 2PKNN [4], JEC-DF [5], and JEC-AF [6]. Finally, the conclusion summarizes the current findings and identifies potential directions for future work.

II. CNN RELATED WORKS

Indeed, the rapid growth of archives of available visual content, such as photo or video sharing websites, has created a need for indexing techniques and multimedia information search, and more specifically images. With image annotations, large collections of images can be indexed and searched in a fast and convenient way. Image Annotation and feature extraction are two topics we will examine in our study, so let's take a closer look at what has been done in these fields before.

A. IMAGE ANNOTATION FEATURES

It is established that all extracted regions in image annotation can be represented by various features including colors, textures, structures, and shapes information. In this study, a diverse set of features characterized each image region to increase the performance of the image annotation algorithm in terms of the shapes extraction, computational cost reduction for identifying most of the suitable features extracted from the training and testing images as underscored below Features Extraction Method. Two categories of features extraction methods such as global (colors, textures and shapes) and local (corners and edges) were used. These features are described hereunder.

1) LOCAL FEATURES

Several descriptors have been developed over the years to describe local features, such as regions, segments, and corners of an image. To extract these local features, numerous algorithms have been used. For example, SIFT (Scale-Invariant Feature Transformation) by Lowe [7]. is a very popular local feature extraction algorithm. Algorithm dependent on high dimension for matching that is invariant to scale and rotation. Herbert introduced SURF, a technique inspired by SIFT and believed to be faster than SIFT, to solve the high-dimensionality matching problem [8]. However, the rotational invariance performance of SURF could be better. Reference [9] presented the Histogram of Oriented Gradients (HOG) algorithm for identifying local features. This algorithm performs better than existing methods of describing local features, capable of categorizing the object appearances and shapes. A critical review of relevant literature on diverse features extraction techniques revealed that no standalone global and local features, have proved insufficient for describing the image. Thus, for all visual aspects of an image to be described precisely, a robust feature extraction framework is needed.

2) GLOBAL FEATURES

Over the last few decades, several feature extraction algorithms have been developed to extract global features (colors, textures, and shapes) from the images. A color strongly correlates with an image's objects, foregrounds, and backgrounds, making colors the most prominent and attractive features. The four most common color representations are the color histogram, color moments [10], color correlogram [11] and color co-occurrence matrix [12]. Two categories of color spaces exist such as linear (YCBCR, CMY, YIQ, RGB, XYZ, and YUV) and nonlinear [13]. Although the color features in Content-Based Image Retrieval (CBIR) approaches Despite their popularity, they are not sufficient to describe an image fully nonlinear [14]. These color feature descriptors have many limitations as well, including lack of perception similarity and lack of spatial information. A texture lacks a proper definition.

After considering colors and shapes of an image, texture is the last thing left to consider [15]. Despite the fact that the texture feature is very beneficial in the photo for CBIR, it is also limited by several factors, such as complexity, accuracy, and noise sensitivity. Many texture-based CBIR systems have been proposed to improve the accuracy of CBIR schemes. Some common texture features extraction algorithms are Markov Random Field (MRF), Edge Histogram Descriptor (EHD) [15], Steerable Pyramid Decomposition (SPD) [16], and Gray Level Co-occurrence Matrix (GLCM) [17]. Many researchers have combined shape with color or texture to enhance the performance of CBIR systems by combining image shape with features like color and texture. Shape features can be extracted from images based on several algorithms, including Multi-Texton Histograms (MTHs) [16], Curvature Scale Spaces (CSSs), and Fourier Descriptors [17]. Generally, the shape feature descriptors are sensitive to stability and translational, scaling, and rotational invariance. To surmount such drawbacks of the existing algorithms, different CBIR systems have been proposed to achieve better accuracy and efficiency, texture, color, and shape are combined [15].

B. IMAGE SEGMENTATION

Most image segmentation techniques used in research primarily focus on the color space of the image. These methods extract image visual features either globally or locally. Global methods analyze the entire image for a set of features, while local methods divide the image into blocks or regions and compute a set of features for each block. This allows images to be represented with object-level features while maintaining spatial information. However, unsupervised segmentation with region features may impact accuracy, as segmentation performance depends on the intended use. Common algorithms for image segmentation include grid-based, clustering-based, contour-based, region growing-based, and statistical model-based techniques. The variance intra-cluster maximization method is a highly efficient image segmentation technique as it selects a global threshold value by maximizing the separation between classes in gray-level images [18].

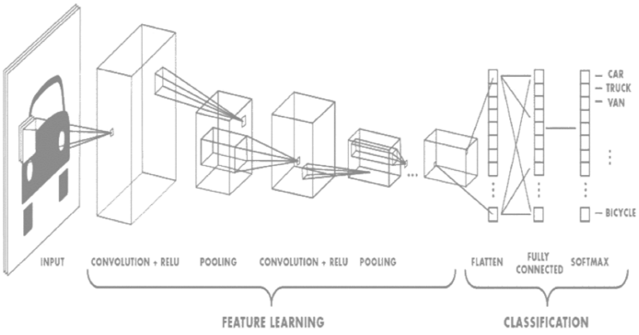


FIGURE 1. The general CNN configuration [27].

C. AUTOMATIC IMAGE ANNOTATION METHODS USING CNN

In this paper, a brief review of the deep learning methods for AIA was conducted. Convolution neural network. Convolutional Neural Networks (CNNs or ConvNets) are a popular type of deep feed-forward Artificial Neural Networks utilized in visual image analysis. These networks are influenced by the visual detection capability of living beings. Variants of CNN architecture, such as LeNet-5, AlexNet, VGG, GoogleNet, and Deep Residual Learning, exist in the literature, but they all share basic components. For example, LeNet-5 has three fundamental layers (convolutional, pooling, and fully-connected) as seen in Figure 1. It represents the input feature representation learned by the convolutional layer, which comprises of multiple convolution kernels for computing diverse feature maps. Each neuron's feature map is connected to a nearby region in the previous layer (known as the neuron's receptive field). The input undergoes convolution with a trained kernel before being processed with a component-wise nonlinear activation function to produce the new feature map. It's important to note that before generating each feature map, all the inputs must share the same kernel and multiple kernels are necessary to produce all the feature maps [19].

The ConvNet architecture features successive conventional layers, which reduce the computational burden and broaden the network's perspective. This is accomplished by reducing the spatial size of the representation and controlling overfitting through the pooling layer, using the MAX operation as shown in Figure 2, which operates independently of the response's depth slice.

The most commonly used pooling layer in CNNs has 2 × 2 filters, which reduces the complexity of an image by down-sampling by two in both width and height, retaining 75% of the activations [19]. CNNs have improved the performance of several computer vision tasks by learning from large amounts of supervised data. Some of the popular CNN-based AIA and retrieval models include the combination of

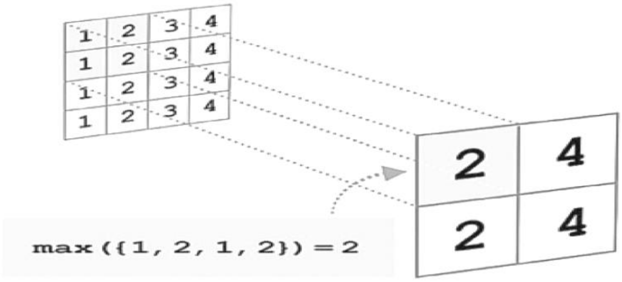


FIGURE 2. The max-pooling operation using 2 × 2 filters [27].

CNN features and semantic extension model (SEM) using AlexNet [20], which was evaluated on standard image annotation datasets. However, its accuracy was limited due to non-uniform tags in the dataset. Another study proposed MVAIACNN for large-scale image annotation using raw images and shallow layers [19], which was evaluated on MIRFlickr25K and NUSWIDE datasets. In multilabel image annotation, the number of labels and their annotations must be fixed and determined by a ranking function. The authors also proposed a CNN-THOP for image annotation and improved the VGG16 architecture for faster convergence and better network structure. Another CNN-based approach was suggested for annotating power grid images, which achieved 94.83% accuracy. The authors also proposed a feature combination technique for image annotation and retrieval and a DL and computer vision-based framework for automatic unlabeled coral image annotation [19]. This framework used a coral classification CNN and validated its accuracy with human experts. The trained coral classifier was applied to analyze the Abrolhos Islands' coral reefs, where a two-year increase in accuracy was observed. However, misclassifications were still prevalent, mainly due to time-related changes in coral reefs and uncertain coral-non-coral boundaries.

III. RESEARCH PROBLEM DEFINITION

Several issues in the existing image annotation systems need to be overcome. Firstly, this operation can be seen as a function to associate the visual information represented by the low-level characteristics (shape, color, texture, and so forth) of the image, wherein the semantic information is characterized by its keywords. Currently, the IA system's major challenge is bridging the semantic gaps between the low-level computational characteristics and human interpretations of the images. The image interpretations include the concepts of various levels of abstraction that cannot simply be matched with the features, which requires additional reasoning with general and specific knowledge of one area. For instance, the images in Figure 3(a) and (b) have the same semantic meaning but completely different appearances. In addition, the images in Figure 3(b) and (c) are visually similar but do not have the same semantic meaning. Based on this description, it became clear that the aforementioned semantic gaps are the main limitations of the existing image classification systems, which need to be resolved.

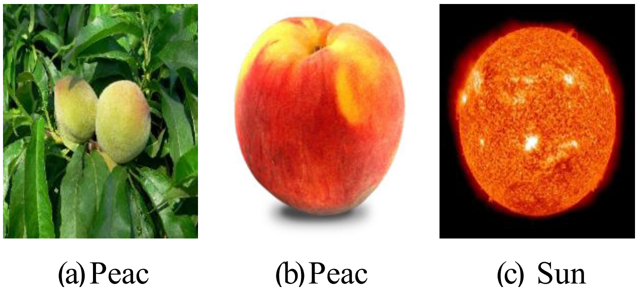


FIGURE 3. Example of the semantic division and gaps.

Secondly, existing solutions need a higher rate of accuracy Which leads to a lack of information in its entirety in circumstances where the created image tags and adversely affects the retrieval image system [18]. Thus, it is important that the image annotation solutions address this issue and to increase accuracy to avoid the drawbacks of a possible empty label set for images.

In addition, existing image annotation solutions suffer low images that can be indexed based on their visual contents to be searched and retrieved later. Although some of these solutions employed several ML, DL, DNNs, ANNs techniques may be prospective to bridge the existing gaps. For an effective image annotation system, decreasing the number of labels in the image is important. Thus, a new scheme is a robust image classification scheme called automated image annotation (AIA).

IV. ANNOTATION APPROACH ARCHITECTURE

Our proposal in this section details the automatic image annotation approach, which presents and describe an image annotation scheme based on deep learning and then describe a new representation of annotation features Our automated image annotation is then achieved by using the cooperative training method.

A. AUTOMATIC IMAGE ANNOTATION (AIA) SCHEME

Automatic Image Annotation (AIA) aims to automatically match an image with a set of keywords chosen from a predefined lexicon. To rephrase, the input is the desired image, and the output is a list of keywords that most accurately characterizes the image. Though computers can quickly and readily calculate the low-level features from colors, textures, and shapes, they need to provide a semantic interpretation of these features, in contrast to humans. Therefore, connecting the dots between the low-level computational features and human interpretation of images is the primary problem in AIA [21], [22]. The AIA has been studied intensively in recent years to find answers to these problems. As a result, many theories have been presented as potential solutions to this issue.

This section offers a comprehensive summary of the AIA design that has been proposed. The envisioned AIA system was implemented in three stages. In the first place, the

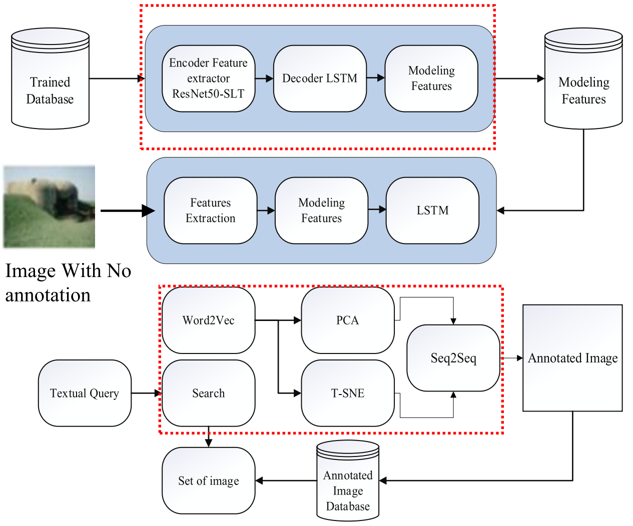


FIGURE 4. **Proposed ResNet50-slt training and testing framework of AIA scheme.**

training was the most important aspect of the system, and a database of labeled images was employed for this purpose. The second step was for the trained system to operate on the unprocessed input and produce the annotated image. Finally, in order to assess the efficacy of the suggested AIA system, image retrieval was finally conducted. The initial training stage used an automatic features extraction with ResNet50 and the standard training database. The feature vector was quickly and generated by the automatic feature extraction procedure thanks to the contextual awareness of the images. Annotating the fresh image required first modelling their features through a learning method, then generating the model for annotation.

The raw image without annotation was used as input in the second stage. Once the annotation model was trained, the next step was to extract the features that would provide the contents' visual qualities. The model created in the first stage was utilized to assign the appropriate semantic label to the image based on its contents, resulting in an image with annotations. The image was then labelled with annotations as a consequence. The images from the annotation stage were used as the database for the third and final stage. The system returned a set of visually-related results in response to the textual inquiry. Annotation made it possible to more quickly and accurately retrieve photos based on their content. Figure 4 depicts the proposed architecture for the AIA system.

The following graphic depicts the proposed method for AIA, which incorporates the feature extraction model paired with Slatelet transform introduced in next section. The suggested system includes the following three stages:

The new features extraction and Annotated images are used for training so that the model may learn to generate annotations automatically given an input image. At this stage, the raw image is analyzed. Using the annotation model that was previously trained, the next step is to extract visual features from the image. An annotated image is produced as a result of the model's use of a trained word2vec language model in conjunction with principal component analysis and t-distributed stochastic neighbor embedding to provide automatic correct semantic labels. Since the annotation is content-based, image retrieval becomes simpler and more accurate in the third phase, when it is utilized to identify a list of relevant photos given an input text. The proposed system architecture for automatic image annotation is depicted in the following figure 4.

B. NEW FEATURES EXTRACTION

The main reason for the IA is the raw image pixels' inability to provide sufficient unambiguous information, thus creating the semantic-level concepts between low-level language and high-level language. the task of annotating images using deep learning can be challenging due to the complexity and diversity of natural images [23]. This implies that most Existing features used for describing images may not be sufficient and none of them can adequately represent the wide variety of images found in nature. Therefore, generating new feature vectors is crucial to avoid semantic gaps, increasing the accuracy while maintaining a higher level of image retrieval. In this research, the anomaly detection schemes utilise the reduced data to further reduce the computational complexity of detection methods. This section highlights the crucial role of feature extraction (FE) in an AIA model, transforming raw images into meaningful features. Image features can be broadly classified into two categories, low-level (shape, color, texture, etc.) and high-level (representing concepts or words). Deep learning has seen substantial progress in computer vision with the aid of extensive imaging datasets. Through deep learning, many features can be extracted across various layers [2], [24], Several deep learning models are recently proposed such as VGG [25], AlexNet [26]. In this study, we employ pretrained ResNet-50-SLT models for extracting deep features. The ResNet-50 designation originates from its ability to process neural networks with 50 layers. Typically speaking, ResNet 50 is a very large neural structure. Its superior performance can be attributed to the many convolutional layers and numerous skip connections that make up the network.

Use ResNet50 to generate 4096-dimensional feature vectors for each image. This is done by passing each image through the ResNet50 network and extracting the output of the final fully connected layer. The 4096-dimensional feature vector represents the high-level features of the image that the network has learned. Perform the slantlet transform on the 4096-dimensional feature vectors. The slantlet transform is a type of mathematical operation that is used to extract features from a signal. It is a type of wavelet transform that is able to extract both spatial and frequency information from a signal. In this case, the signal is the 4096-dimensional feature vector, and the slantlet transform is used to extract additional features from the vector.

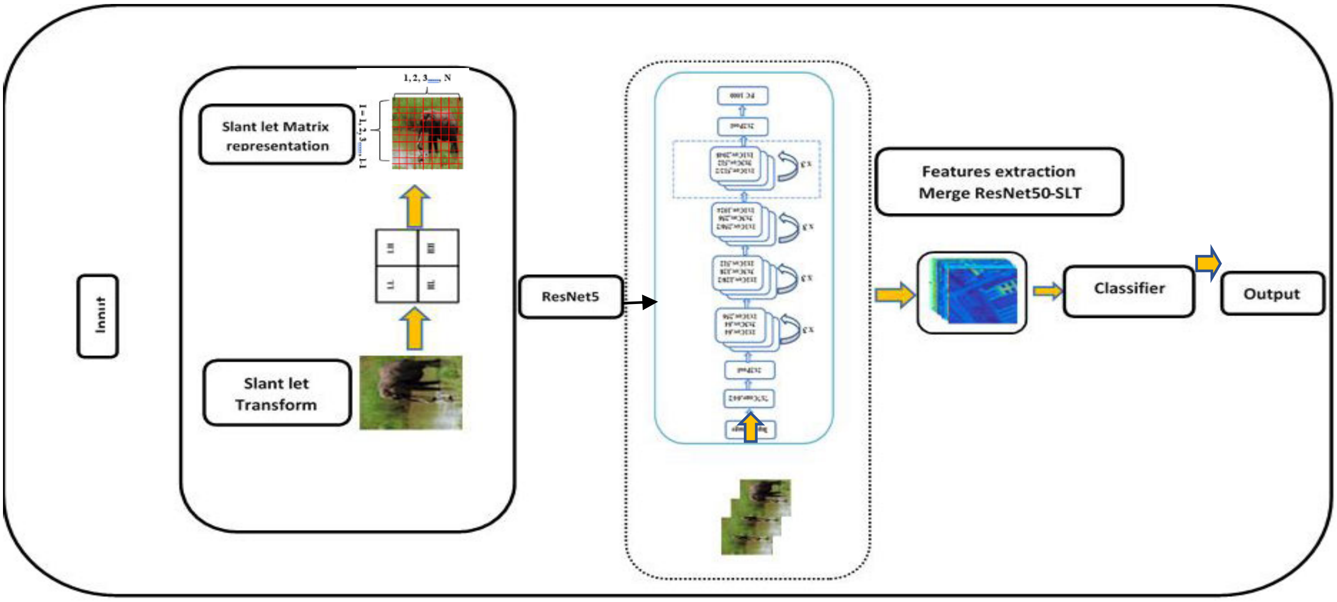

Innut
Slant let Matrix representation
1, 2, 3___ N
Slant let Transform
ResNet5
Features extraction Merge ResNet50-SLT
Classifier
Output

FIGURE 5. New feature extraction approach.

The resulting feature vectors from the slantlet transform are then used as input to the LSTM for decoder. The LSTM can use these features along with its memory of past inputs to generate a caption for the image. Overall, the figure 5 shows the suggested feature extraction approach that is used to extract rich and robust new feature representation from images, by first generating a 4096-dimensional feature vector for each image using ResNet50 and then performing the slantlet transform to extract additional features from the vectors.

1) SLT

The SLT (Slantlet Transform) is highly regarded for its orthogonality and two zero moments, making it an effective method for enhancing time localization. DWT implementations that have a scale dilation factor of 2 utilize different filters compared to those with different implementations. In 2D SLT, an image is broken down into four parts: LL (Low-Low), LH (Low-High), HL (High-Low), and HH (High-High). The LL component holds the overall image information, while the LH, HL, and HH components contain information about the edges, contours, and other details in the image. For improved results in subsequent operations, it is recommended to ignore the small coefficients in the image that hold no useful information.

Complete Slantlet transform can be represented in matrix format by

$$S = \boldsymbol{SLT}^{N} s \boldsymbol{SLT}_{N}^{T} \tag{1}$$

The matrix describes the Slantlet transform of a 2D signal (s) and the matrix (SLTN) represents the Slantlet transformation of the original signal (s). The size of s, S, and SLTN is the same (N x N). The Slantlet transformation is applied to the original signal, resulting in a coefficient matrix that can be represented as follows

$$S = \begin{bmatrix} S_{0,0} & S_{0,1} & \cdots & S_{0,N-1} \\ S_{1,0} & S_{1,1} & \cdots & S_{1,N-1} \\ \vdots & \vdots & \ddots & \vdots \\ S_{N-1,0} & S_{N-1,1} & \cdots & S_{N-1,N-1} \end{bmatrix} \tag{2}$$

In the proposed method, matrix multiplication will be utilized to calculate the SLT coefficients for image blocks. The matrix (S) will be divided into four sub-bands (LL, HL, LH, and HH) based on the SLT coefficients obtained from the matrix multiplication process [27].

2) SLANTLET MATRIX REPRESENTATION

The proposed method involves transforming each block in the spatial domain using the SLT matrix, as described in equation (3). The size of the spatial domain block and the SLT matrix is the same. Then, a carrier subband is selected (either the HL subband or the LH subband) and the mean value of this subband is calculated. This step is repeated for all blocks in the spatial domain. The next step is to find the maximum absolute mean value (mmax) and use it to set the threshold. The threshold is the smallest integer greater than mmax. The shift value is set to be greater than the threshold. In order to ensure reversibility, a trial-and-error process is used to find the suitable shift value. This is done by first trying the shift value equal to (mmax + 1) and checking for errors in the recovered image. If errors are found, the shift value is increased.

In most cases, the shift value equal to (mmax + 1) is sufficient. This process aims to control the quality of the extracted image features. Increasing the shift value degrades the visual quality, so the aim is to find the smallest shift value that can be used while still preserving image quality. Slantlet

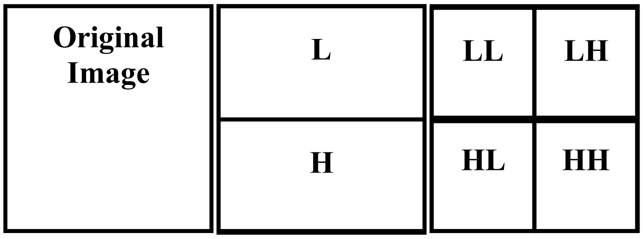


FIGURE 6. Proposed ResNet50-slt training and testing framework of AIA scheme.

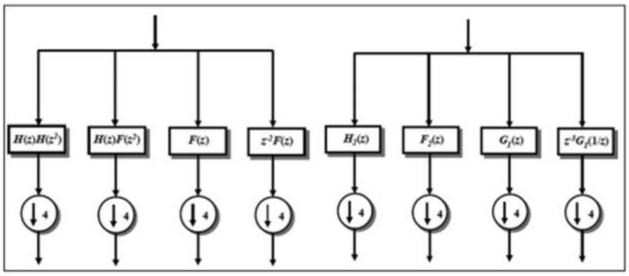


FIGURE 7. Two-scale filterbank and an equivalent structure.

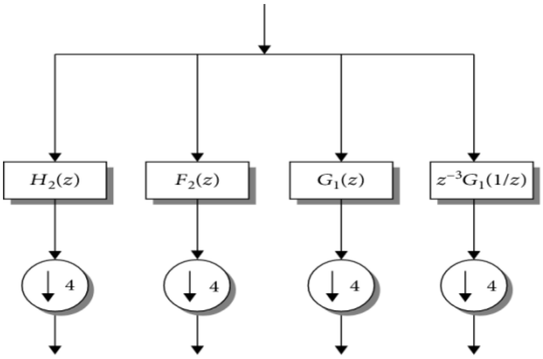


FIGURE 8. Final Two-scale filterbank structure.

transform, first employed by Selesnick to study nonstationary signals, is an improved orthogonal DWT variant with two zero moments and higher time localization; we pair it with ResNet50 to extract stronger statistical characteristics inside the images. The typical approach to DWT is a filter bank iteration that is tree-based. The Slantlet transform cues from the parallel filter bank layout of a DWT implementation's equivalent. The "Slantlet" filter bank, which operates in parallel with DWT, also uses a framework in which the product form of fundamental filters is used. On the other hand, since SLT doesn't treat component filter branches as a product type, it has more independence. When using SLT, a group of filters will be provided, each with a length that is a power of 2. This allows for regular output from the analysis filter bank and reduces the required number of samples to support an increase in the analysis by as much as a third.

$$g_i(n) = \left\{ \begin{array}{l} a_{0,0} + a_{0,1}n, \text{ for } n = 0, \cdots, 2^i - 1 \\ a_{1,0} + a_{1,1}n, \text{ for } n = 2^i, \cdots, 2^{i+1} - 1 \end{array} \right\}$$

$$h_i(n) = \left\{ \begin{array}{l} b_{0,0} + b_{0,1}n, \text{ for } n = 0, \cdots, 2^i - 1 \\ b_{1,0} + b_{1,1}n, \text{ for } n = 2^i, \cdots, 2^{i+1} - 1 \end{array} \right\}$$

$$f_i(n) = \left\{ \begin{array}{l} c_{0,0} + c_{0,1}n, \quad \text{ for } n = 0, \cdots, 2^i - 1 \\ c_{1,0} + c_{1,1}n, \text{ for } n = 2^i, \cdots, 2^{i+1} - 1 \end{array} \right\} \tag{3}$$

The Slantlet transformation can be handled more precisely if we switch to a more generalized representation of Figure 7 The signal analysis cannot proceed without properly functioning filters at scale I. As a result, the SLT filter bank uses channels, or channels in total, to process signals of size l. (l). Each filter is downsampled after the low pass filter and then combined with its neighbor. The signal is downsampled initially and then again by its time-shifted form at I = 1, 2, 3…., l-1 samples earlier. As filter and linear form fragment implementations, the following symbols are used below in figure 8 The general equation of SLT filters.

SLT for AIC images took consideration of both factors. Figure 4.5 show that the length of the input signal must be a power of two or greater if the filter lengths in an SLT filter bank are to be a power of two as well. Second, a matrix of transformations is constructed.

When using 2D SLT decomposition, creating an approximation and precise version of an image is usual practice. In the approximation part, we have one low-frequency subband (LL); in the detailed part, we have three high-frequency subbands (LH, HL, and HH), where H and L represent the high and low frequency bands, respectively. The artist's details are hidden in the original image's low-frequency subband component (LL). However, the LH, HL, and HH subbands keep the image's edges, contours, and other fine elements intact. Details in the image with high coefficients stand out against the background noise with low coefficients. Therefore, it is preferable to avoid using such small coefficients wherever possible. This study used the SLT to extract statistical information from AIC images in the spatial and neutrosophic domains.

V. ResNet-50 PROPOSED

The ResNet 50 is a deep learning model introduced in 2016 by He et al. with 50 layers. It has a similar structure to the VGG networks, with 3 × 3 filters in the convolutional layers and an input size fixed at 224 × 224. The design of the ResNet 50 model is straightforward, with all layers having the same number of filters resulting in the same output size. When the output size from the convolutional layer is halved, the number of filters is doubled to maintain the same computational complexity per layer. The model ends with an average pooling layer and a 1000-way fully-connected layer with a softmax activation function.

In comparison to the VGG networks, the ResNet 50 model has fewer filters and a lower computational complexity. There are other variations of this model, including ResNet101 and ResNet152. The configuration of the layers in the ResNet 50 network is shown in Figure 9.

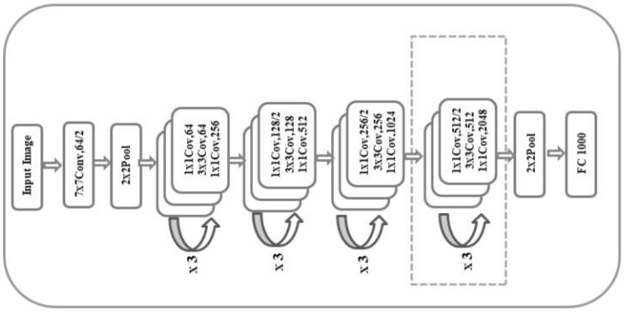


FIGURE 9. ResNet50 model architecture.

The amount of (configurable block group) blocks at each level varies widely and depends entirely on the ResNet architecture employed. For this example, we've decided to use 50 layers, and you can get a better look at them in the figure we've included. As a means of inspecting a more comprehensive structure. The ResNet-50 architecture blocks Fully-connected layer Different versions of the ResNet architecture use a varying number of (configurable block group) blocks at different levels, as mentioned in the figure above. Cfg0 This block contains 1 Conv Layer and 2 Identity Layers. A kernel constraint improves numerical stability, which ensures that all weights are normalized at constant intervals. Between 2 subsequent layers, we also include a Batch Normalization layer. Cfg1 Block This block contains 1 Conv Layer and 2 Identity Layers. This is similar to the Cfg0 blocks, with the difference mainly being in the number of out channels in the Conv and Identity layers being more. This block contains 1 Conv layer and 5 Identity layers. Cfg2 This is one of the more important blocks for ResNet as most versions of the model differ in this block-space. Cfg3 This block contains 1 Conv Layer and 2 Identity Layers. This is the last set of Convolutional Layer blocks present in the network. Classifier Block This block contains an Average Pooling Layer, a Dropout Layer and a Flatten layer. At this block, the feature map is finally flattened and pushed into a Fully Connected Layer which is then used for producing predictions. Finally, a Softmax activation is applied to generate logits/probabilities. As gradient super-highways, skip connections prevent the issues above from occurring and keep the flow from being significantly affected. ResNet is largely responsible for spreading the concept of Skip Connections.

The purpose of skip connections, as the name implies, is to skip over certain levels of a neural network and send the output directly to the layer below, as shown on Figure 10 As a result, the problem of Degradation is much better now. When the skip link is included, the leftover block takes on a new appearance:

The layer's output is summed and sent to the next layer via skip connections, eliminating the need for additional parameters in the processes. This facilitates not only answers to problems associated with picture classification, but also segmentation, key-point recognition, and object detection.

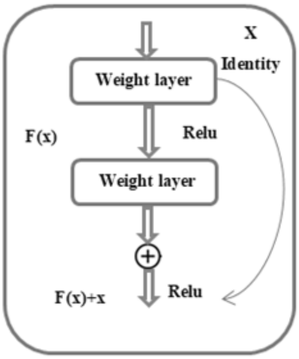


FIGURE 10. Skip connections in ResNet50.

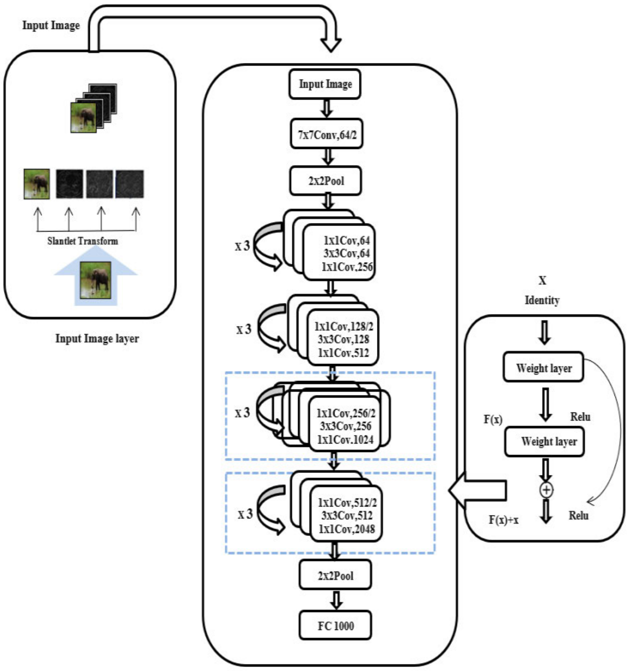


FIGURE 11. The ResNet50-SLT architecture.

VI. COMBINATION OF ResNet50 AND SLT

Various regional, segmental, and angular descriptors have been created over time to better characterize certain image elements. Many different approaches have been developed to extract these neighborhood characteristics.

our approach can be defined local properties, and it can classify objects based on their external look and geometric structure. The image cannot be described using either global or local features, as was found by a thorough analysis of the literature on feature extraction approaches. Accordingly, a powerful feature extraction framework is required to represent all visual features of an image here and accurately describe the Combination details Resnt50-slt is shown on figure 11.

In this paper, we will develop the mathematical details of the ResNet50-SLT architecture. Consider H(x) as an underlying mapping to be fitted by a few layers (not necessarily the entire net), where x indicates which inputs to the first layer constitute the residual functions, i.e., H(x) − x (assuming that the input and output are of the same dimensions) (assuming that the input and output are of the same dimensions). Therefore, instead of assuming that stacked layers will approximate H(x), we allow them to do so with the explicit goal of approximating a residual function: F(x): = H(x) − x. This modification of the original function yields F(x+x). After every few layers of stacking, we implement residual learning. An elementary building block is considered, with the formula

$$y = F\left(x, W_i\right) + x \tag{4}$$

The input and output vectors for the layers under consideration are denoted by x and y, respectively. The desired residual mapping is represented by the function

$$F\left(x, w_j\right) \cdot F = W2\sigma\ \left(W1_x\right)\left(W1_x\right) \tag{5}$$

where stands for the ReLU and the biases have been left out of the notation for clarity. Using a fast-track connection and element-by-element addition, we get the result F + x. In this case, we use the second nonlinearity in the sum (i.e., (y)). The simplified relationships of Eqn. (5) add neither new parameters nor complexity to the computation. It is essential that x and F have the same dimensions for Eqn to hold (1). If not (when switching input/output channels, for example), we can use the linear projection Ws by the short-cut connections to bring the dimensions into alignment:

$$y = F\left(x, W_i\right) + WS_x \tag{6}$$

For Eqn, we can also use a square matrix Ws. It is possible to represent several convolutional layers with the function F (x, Wi). Elements from two feature maps are added together, one channel at a time. Consequently, we need to find values and such that the desired -scale filterbank is orthogonal with two zero moments. By expressing the orthogonality and moment conditions as a multivariate polynomial system, we obtain the following solution for and:

$$\begin{aligned}
m &= 2^i \\
u &= 1/\sqrt{m} \\
v &= \sqrt{\left(2m^2+1\right)/3} \\
b_{0,0} &= u \cdot (v{+}1)/(2m) \\
b_{1,0} &= u - b_{0,0} \\
b_{0,1} &= u/m \\
b_{1,1} &= -b_{0,1} \\
q &= \sqrt{3/\left(m \cdot \left(m^2-1\right)\right)}/m \\
c_{0,1} &= q \cdot (v - m) \\
c_{1,1} &= -q \cdot (v + m) \\
c_{1,0} &= c_{1,1} {\cdot} (v{+}1 - 2m)/2 \\
c_{0,0} &= c_{0,1} \cdot (v{+}1)/2
\end{aligned} \tag{7}$$

Afterwards, thresholding is performed Afterwards, thresholding is performed, where a sufficient threshold is selected to eliminate the SLT coefficients that lack significant energy. The features of various PQ disturbances are extracted using the coefficient of at each decomposition level as exhibited in Algorithm 1.

As an example, it's easy to see how going from 3 channels to 32 channels, using a simple 7 × 7 kernel convolution layer, increases the number of parameters by 4736. The training of the model becomes even more difficult as the number of layers grows. So, to train effectively, more memory and computational capacity are needed.

Algorithm 1 ResNet50-Slantlet Transform

```
Input: image slantlet transform
Output: features extraction
convert image to gray
convert image to four sub domains
compute s = 1 / sqrt (2) * np. array ([1, 1], [1, -1])
For i in to 2, n+1
        a. compute b=1 / sqrt (1 + 4 * a*2)
                b. a = 2*b*a
                c. compute points q1, q2, q3, q4
                d. If i == 2:
                i. calculates b1 and b2
                  ii. s = (1 / sqrt (1)) * con ([b1, b2]) @ z
                  e. else:
                  calculate matrix b1, b2, b3 and b4
          s = (1 / sqrt (2)) * con ([b, 1b2, b3, b4]) @ z
         feat(conv) = (feat/ (mxn)) + s
End
End
```

Accuracy is a significant problem because of the increased complexity of neural networks used by modern networks. Parameter counts in these models expand exponentially as their depth grows. CNNs, also known as convolutional neural networks, have convolutional layers as its foundation.

VII. ResNet50–LSTM MODEL

The motivation behind Automatic Image Annotation (AIA) is to improve the accuracy and efficiency of existing image annotation systems. This is because existing solutions have low accuracy rates and lack of information in image tags, making image retrieval systems less effective. To overcome this, AIA uses a combination of word2vec, PCA, and T-SNE to increase the number of labels in an image and improve accuracy. The algorithm detects neighboring words to annotate more words within an image. The system utilizes the ResNet-50 and an LSTM, with the final fully connected layer of the ResNet-50 model being used as the input to the LSTM. The goal of AIA is to create an image annotation system that accurately captures information within an image and provides relevant labels for a more efficient image retrieval system.

The proposed model, ResNet50-LSTM, combines the advantages of convolutional neural networks (CNN) and recurrent neural networks (RNN). The model consists of two main components: the first component is made up of convolutional and pooling layers, which perform mathematical operations to develop the features of the input data, while the second component is made up of LSTM and dense layers, which exploit the generated features. The ResNet-50 model is a CNN that is trained to extract features from an input image. It consists of several layers, including convolutional, max pooling, and fully connected layers. The output of the final fully connected layer is a feature vector that represents the image, known as the image embedding. The LSTM is a type of RNN that is trained to predict a sequence of labels based on a sequence of inputs. The proposed ResNet50-LSTM model consists of five convolutional layers with 64, 128, 256, 512, 1024, and 2048 filters of size (3), followed by two max pooling layers with size (2), an LSTM layer of 2048 units, a dense layer of 512 neurons, and an output layer of one neuron. An illustration of the proposed model's architecture is shown in Figure 12. To merge ResNet-50 with an LSTM, utilized the output of the final fully connected layer of the ResNet-50 model as the input to the LSTM. In this case, the input to the LSTM would be the embedding of the image, and the output of the LSTM would be the predicted label or labels for the image.

Mathematically, this can be represented as follows: Let X be the input image and F be the output of the final fully connected layer of the ResNet-50 model. F = ResNet50(X) where F is the image embedding. Let Y be the output label or labels for the image and L be the LSTM model. Y = LSTM(F) where Y is the predicted label or labels for the image. The above steps can be represented in one equation as Y = LSTM(ResNet50(X)).

VIII. NEW WORD EMBEDDING PCA AND T-SNE

The dimensionality of a dataset can be decreased in a non-linear fashion using a technique called stochastic neighbour embedding. Combining principal component analysis (PCA) and stochastic neighbor embedding (SNE) improves data presentation. Simply said, SNE is an attempt to express a high-dimensional data matrix (X) in a low-dimensional space (Y). Since the points are placed by being drawn to similar points and repelled by different ones, the process produces a clustering approach. This is done by describing the high-dimensional Euclidean distance as a probability pij and then match it to a probability qij from a low-dimensional space. Initially, the positions in the low-dimensional space are initialised at random. In contrast to the non-symmetric probability used in the initial SNE implementation, symmetric SNE is a more recent variant. Through the use of symmetric probabilities, symmetric SNE is able to pair high-dimensional pij with low-dimensional qij values. Two data points' pij and qij can be determined using the following equations for symmetric SNE:

$$\begin{aligned}\mathrm{p}ij &= \mathrm{exp}(- \parallel \mathrm{xi} - \mathrm{xi} \parallel 2/2\sigma 2)/\sum \mathrm{k6}\\ &= \mathrm{lexp}(- \parallel \mathrm{xk} - \mathrm{xl} \parallel 2/2\sigma 2) * \mathrm{W}\end{aligned} \tag{8}$$

and

$$\begin{aligned}\mathrm{qui} &= \mathrm{exp}(- \parallel yi - yj \parallel 2)/\Sigma k66\\ &= 1\mathrm{exp}(- \parallel xk - y \parallel\parallel 2)\end{aligned} \tag{9}$$

where 2 is the variance of the Gaussian curve, which is set to 1/2 for every qij since the density is less likely to change considerably from one location to another in that instance. Symmetric SNE seeks to accurately represent each pij by modeling it with a corresponding qij. Keeping to the original in such a way means retaining as much of the local form as feasible, as was previously mentioned. Particularly, symmetric SNE would rather not model a small qij with a high pij, while the converse is less of an issue. The following expression gives the non-symmetric Kullback-Leibler divergence cost function that captures this property.

$$\mathrm{C} = \mathrm{KL}(\mathrm{P} \parallel \mathrm{Q}) = \sum \mathrm{I} \sum \mathrm{jpijlog\ pij/qi} \tag{10}$$

For every I j, the Gaussian curve's variance 2 must be stated. The symmetric SNE "perplexity setting" is a tool that can be used to achieve this goal. The perplexity setting is a global parameter that stands in for the effective number of neighbors each point in the algorithm must take into account. By adjusting the user-specified perplexity, one may calculate the variance of the Gaussian curve for pij as: Perplexity (Pi)=2Pi, where Pi is the Shannon entropy corresponding to:

The optimization procedure aims to reduce the overall Kullback-Leibler divergence cost. One method for accomplishing this is through (random) gradient descent on the relevant gradient. Formulas for the gradient and gradient update are as follows

$$\begin{aligned}\mathrm{xi} &= 4j(\mathrm{piigiij})((\mathrm{xiyi})\text{ and }\mathrm{Y(t)} = \mathrm{Y(t1)} + \mathrm{C/Y}\\ &\quad + \mathrm{(t)(Y(t1)Y(t2)respectively.}\end{aligned} \tag{11}$$

where is the learning rate and (t) is the momentum at the current iteration t (first specified as a parameter). As there is not enough low-dimensional space to describe all the pairwise high-dimensional relations, SNE tends to cluster data points in the middle of the graph. T distributed stochastic neighbour embedding is a non-linear dimension reduction method that, like SNE, solves the over crowdedness issue. Both SNE and t-SNE use a stochastic method to reduce data from a high-dimensional space to a lower-dimensional one, hence they function similarly. Instead of the normal distribution used by SNE, a student t-distribution with one degree of freedom is employed by t-SNE to represent the qij's. Because the t-distribution has wider tails, this modification helps alleviate the congestion issue. Thus, t-SNE is superior than other methods for modelling big paired high-dimensional distances by huge low-dimensional distances. Thus, it can be written as

$$\mathrm{qij} = (1+ \parallel \mathrm{yiyj} \mid 2)\mathrm{1k6} = 1(1+ \parallel \mathrm{ykxl} \parallel 2) \tag{12}$$

Additionally, this modification alters the shape of the gradient function, making optimization simpler. Considering how closely connected it is, the updating strategy can still be

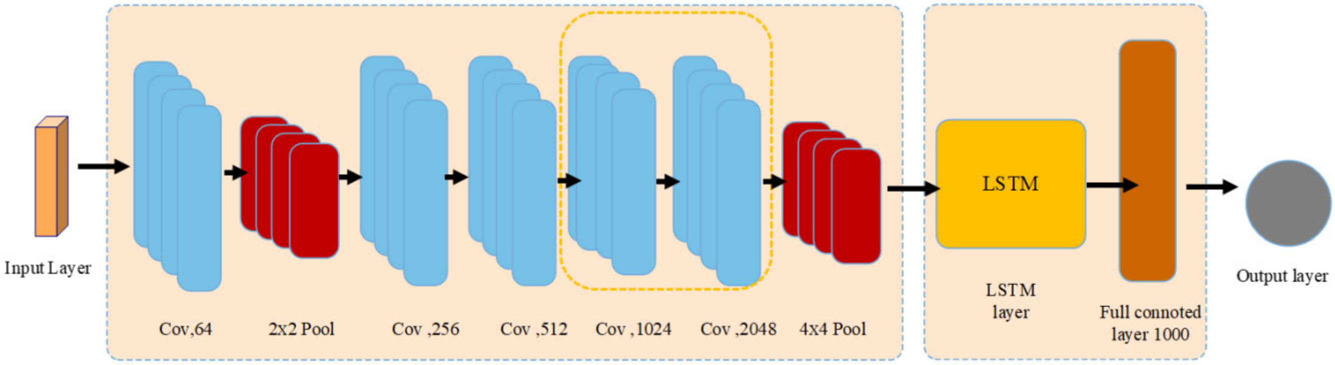


FIGURE 12. Proposed ResNet-50–LSTM model architecture with Fifth convolutional layers, a two-pooling layer, a LSTM layer, a fully connected layer and an output layer.

applied. For T-SNE, this translates to a change in the gradient of

$$Cxi = 4j(piicij)(xiyj) = (1+ \parallel xiyj \parallel 2) \tag{13}$$

Momentum and learning rates are capped. Exaggerating the significance of early pij's can be useful as well. T-SNE is a viable approach to displaying larger datasets seen in real-world research, and it performs well when used to visualization tasks. The t-sne technique is a non-linear approach to reducing the number of dimensions. The algorithm in T -SNE computes the degree of similarity in both high- and low-dimensional spaces. The next step involves utilizing an optimization technique, such as the gradient descend approach, to reduce the gap between the two spaces' levels of similarity as much as possible. Here are a few characteristics of T-SNE.

The T-SNE algorithm is used to reduce the dimensionality of high dimensional data points. It's a non-linear method that adapts to the data and performs different transformations in different regions, which makes it a very flexible algorithm. The T-SNE algorithm is often used in finding structures where other dimensionality reduction algorithms fail. The model uses a given image feature and a caption prefix to generate a new caption word by word. Initially, each caption contains only the artificial start sequence. The image feature vector and the caption are passed to the model. The model then predicts the word from the vocabulary with the highest probability of following the given caption prefix and image feature. The predicted word is appended to the caption and passed back to the model in an iterative process. an image using the proposed model terminates in two ways. The first stopping criteria is when the model predicts the artificial end sequence as the next word. In this case, the caption generation process ends and the final caption is returned. The second stopping criteria is a pre-defined upper bound on the caption length. If the length of the generated caption exceeds the pre-defined limit, the caption generation process ends and the final caption is returned. The pseudocode in Algorithm 2 presents an illustration of the process:

IX. SENTENCE GENERATION

A Seq2Seq model is used for image annotation by converting the input image features into a fixed-length context vector and then using this context vector to generate a sentence describing the image. The input image features are first preprocessed to have a unified length and are then transformed into a word vector representation. The encoder component of the model processes the input image features to produce a context vector, which summarizes the information in the input sequence. The decoder component then takes this context vector as input and generates the output sequence, which is a sentence describing the image. The model uses an embedding layer to convert the input data and the output into 2D arrays with dimensions (sequence_length, vocab_size).

Input(t)

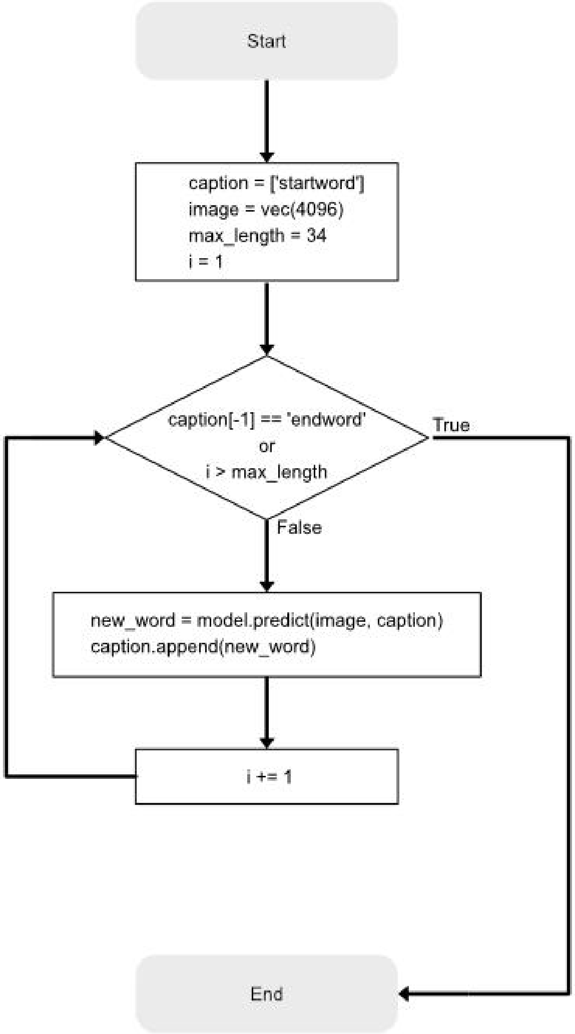
$$= \begin{array}{l} \\ \textbf{Black} \\ \textbf{Dog} \\ \textbf{Happy} \\ \textbf{PlayGrass} \\ \textbf{on} \\ \\ \end{array} \left[\begin{array}{cccccccc} 1 & 32 & 2 & 0\ldots\ldots & 0 & 0 & 0 & 0 \\ 123 & 56 & 3 & 34\ldots.. & 143 & 345 & 0 & 0 \\ 3 & 22 & 4000 & 58\ldots\ldots & 760 & 501 & 0 & 0 \\ 3 & 2 & 941 & 1\ldots\ldots & 0 & 0 & 1 & 0 \\ 888 & 331 & 184 & 1\ldots\ldots & 1 & 0 & 0 & 0 \\ 198 & 270 & 76 & 0\ldots\ldots & 1 & 0 & 0 & 10 \\ 45 & 78 & 0 & 0\ldots\ldots & 1 & 0 & 0 & 0 \end{array} \right]$$

MAX_LEN: to unify the length of the input sentences VOCAB_SIZE: to decide the dimension of sentence's one-hot vector EMBEDDING_DIM: to decide the dimension of Word2Vec with PCA-T-SNE. algorithm that can be used to convert tag words to sentences in image annotation Figure 14 illustrate is a sequence-to-sequence (Seq2Seq) model with an attention mechanism. The basic idea of this algorithm is to encode the input sequence of tag words that take from PCA-T-SNE into a fixed-length vector representation, and then decode this representation into a sequence of words that form a coherent sentence.

Algorithm 2 The Proposed AIA Pseudocode

```
Input: Image Set, the feature set F of Image Set, label set W
Output: Prediction results W' of AIA
For an untagged image feature Fi do
        For image feature $F_I$ In F do
            $$d\left(F', F_i\right) = \sqrt{\sum_{l=1}^{4096}\left(F^{ll} - F_i'\right)^2}$$
        end for
    sort d (F, F) in descending order
    get the neighbourhood feature group $N(I^i,K)$ according to Word
    get the tags $Cl\ (I^i)$ of each image in $N(I^i,K)$
    For w in $Cl\ (I^i)$ do
        For $F^z$ in N (I, k) do
                If w belongs to the image whose feature is $F^z$ then
                    $$\mathrm{smr}\left(F', F^z\right) = \frac{1}{1 + \exp\left(\theta di\left(F', F^4\right)\right)}$$
                    $$p\left(w \mid I'\right) = p\left(w \mid I'\right) + \mathrm{smr}\left(F', F^z\right)$$
        end for
    end for
    sort $\mathrm{smr}\left(F', F^z\right)$ in descending order
        $$a = \frac{\sum_{l=1}^{m} x_l \times smr\left(F', F^L\right)}{\sum_{l=1}^{mm} \mathrm{smr}\left(F', F^L\right)}$$
            For w in $Cl\ (I^i)$ do
                For $w_i$, In W
                    $$w_i \leftarrow \varphi\left(sim\left(w_i, w\right)\right) \geq \partial$$
                    $$addw_itocle\left(I'\right)$$
                    $$p\left(w_i \mid I\right) = p(w \mid I)\cdot\varphi\left(sim\left(w_i, w\right)\right)$$
                end for
        end for
            $$p\left(w_i' \mid l\right) = p(w \mid l) \cup p\left(w_i \mid l\right)$$
            $$sortp\left(w_i' \mid l\right)\ \mathrm{inascendingorder}$$

                Select the top-a $p\left(w_i' \mid l\right)$ and finding the corresponding word
                Add $w_i$, In W
                end for
                Output Prediction results W'
```



```
caption = ['startword']
image = vec(4096)
max_length = 34
i = 1
```

```
caption[-1] == 'endword'
or
i > max_length
```

```
new_word = model.predict(image, caption)
caption.append(new_word)
```

```
i += 1
```

FIGURE 13. Model prediction flowchart.

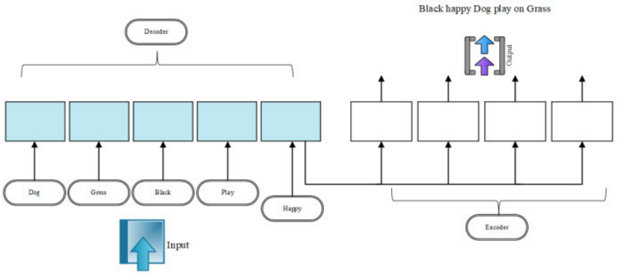


FIGURE 14. Model Seq2Seq for sentence generation.

The decoder, which can also be a LSTM network, takes the fixed-length vector representation generated by the encoder and generates the output sequence (in this case, the caption for the image). The Seq2Seq model aims to maximize the likelihood of the generated caption given the input image and the learned parameters of the model. The final goal is to produce a caption that is descriptive, accurate and coherent with the image content.

$$\begin{array}{c}\textbf{Black Dog Happy PlayGrass on}\\ \text{output(S)} = \begin{bmatrix} 1 & 0 & 1 & 0\ldots\ldots & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1\ldots.. & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1\ldots\ldots & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1\ldots\ldots & 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1\ldots\ldots & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0\ldots\ldots & 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0\ldots\ldots & 1 & 0 & 0 & 0 \end{bmatrix}\end{array}$$

The Seq2Seq model with an attention mechanism can be seen as a combination of the encoder and decoder, where the encoder processes the input sequence to produce a fixed-length vector representation, and the decoder uses this representation to generate the output sequence. The attention mechanism allows the decoder to focus on specific parts of the input sequence while generating the output, which can lead to improved accuracy. The general pseudocode for the Seq2Seq model with an attention mechanism is shown in Algorithm 3 and can be summarized as follows:

1) The input sequence is passed through the encoder, which is typically a ResNet50 or LSTM network, to produce a fixed-length vector representation.
2) The decoder takes this fixed-length vector representation as input and generates the output sequence word by word.
3) The attention mechanism allows the decoder to focus on specific parts of the input sequence when generating each word in the output sequence, improving the accuracy of the model.

Algorithm 3 The Generate Sentence of AIA

```
Input: tag words sequence (t1, t2, . . . , tn)
Output: sentence (s1, s2, . . . , sm)
Encode the input sequence: h = ResNet50 (t1, t2, . . . , tn)
Initialize the decoder's hidden state: s = LSTM.init_state(h+t)
Initialize the attention weights: alpha = attention.
init_weights(w)
Initialize the output sequence:
output = [s]
For each target word in the output sequence:
a. Compute the attention weights:
  alpha = attention (h, s)
  b. Apply the attention weights to the input:
  c = h ∗ alpha
  s = decoder (s, c)
  d. Generate the next word in the output sequence:
  e. Word vectors representation 4096
  word = output layer(s)
  f. Add the word to the output sequence:
  output. Append(word)
  end for
  Return the output sequence as a sentence.
```

The Seq2Seq model with an attention mechanism is a powerful tool for image captioning. It takes in an image and outputs a sentence describing the image. The input image is first passed through a CNN to extract features. These features are then processed by the encoder, which is typically a ResNet50-LSTM network. The encoder produces a fixed-length context vector that summarizes the information in the image features. The decoder, also typically an ResNet50 network, then takes the context vector as input and generates the output sentence word by word, using the attention mechanism to focus on specific parts of the image features as needed.

TABLE 1. Each database contains detailed information.

database	Number of images	vocabulary size	Train size	Test size	Words per image	Images per word
Corel-5K	5 000	260	4 500	500	3.4	58.6
ESP-Game	20 770	268	18 689	2 081	4.7	362.7
Flickr8k	8000	220	7000	1 000	3.9	347.7

X. DATASET

In the experiments, three well-known image annotation databases were used: Corel-5K, ESP-Game, and Flickr8k. The Corel-5K database is a popular database for image annotation and search, and has a vocabulary of 260 keywords that were used for both training and testing. This database consists of 4,500 training images and 500 testing images, divided into 50 categories with 100 images in each category. Each image is annotated with 1 to 5 keywords with an average of 3.4 keywords per image.

The ESP-Game database was used in the experiments, with a subset of 20770 images. This subset includes 18689 images for training and 2081 images for testing, with a vocabulary of 268 keywords. The images are annotated with an average of 4.7 keywords per image.

The Flickr8k database has a collection of around 8,000 natural images that come with a vocabulary of 220 keywords, on average 5.7 keywords per image. It comprises of 8,000 training images and 1,000 test images. A comprehensive examination of the database can be found in Table 1.

Table 2 displays a visual representation of the sample images and their annotations taken from the Corel-5k, ESP-Game, and Flickr8k databases that were used in the experimental section. The table provides a visual representation of the images and the keywords or annotations associated with each image. For instance, the second image in the Corel-5k database is depicted with the annotations "sky," "jet," and "plane." This provides a clear idea of the type of images and annotations included in each of the datasets.

XI. PERFORMANCE EVALUATION

The quality of an image annotation system can be evaluated in two main categories: annotation measures and per-word measures. Annotation measures evaluate the overall performance

TABLE 2. Examples of images from test bases.

Datasets	Image annotation	**Image annotation**	**Image annotation**
Corel-5k			
	sky, sun, clouds, tree	sky, jet, plane	bear, polar, snow,
Flickr8k			
	grandstand, lawn, player, roof, round, stadium	helmet, jean, lamp, man, sweater	city, house, roof, sky, valley, view
ESP-Game			
	round, stone, green, sky, grass, man, bunker, building, concrete	stone, white, cow, dirt, tail, bull, grass,	pink, silver

of the system on a set of images, while per-word measures evaluate the performance of the system on a word-by-word basis. Some of the metrics used in the literature for evaluation of image annotation systems include accuracy, recall, precision, F1 score, and mean average precision (MAP).

These metrics provide different aspects of the system's performance and can be combined to give a more comprehensive evaluation of the system. Accuracy measures the percentage of correctly annotated images. Recall measures the proportion of annotated images that are correctly retrieved. Precision measures the proportion of retrieved images that are correctly annotated. F1 score is a balance between precision and recall and provides a single number that summarizes the overall performance of the system. MAP computes the average precision of the system by averaging the precision scores of each image in the test set. In conclusion, various metrics can be used to evaluate the quality of an image annotation system, and it is important to choose the appropriate metrics that best align with the goals of the particular application.

A. PRECISION AND RECALL

Precision can be defined as the proportion of correctly annotated keywords among all the keywords predicted by the model. It measures the accuracy of the image annotation system in correctly annotating the keywords. Precision is calculated as the number of correctly annotated keywords (m3) divided by the total number of predicted keywords (m2). Precision provides a measure of the relevance of the keywords generated by the system [36].

$$P_e = \frac{m_3}{m_2} \tag{14}$$

Precision measures the accuracy of the model in terms of the number of correct keyword annotations in the image, in comparison to the total number of annotations made by the model. In other words, precision gives an idea of the percentage of keywords that are correctly annotated by the model, out of all the keywords that the model annotates [37].

$$R_e = \frac{m_3}{m_1} \tag{15}$$

B. F-MEASURES

The F-Measure is a weighted harmonic average that combines recall and accuracy, which is given by [38]:

$$F_\alpha = \frac{\left(1+\alpha^2\right)(PR)}{\alpha^2 P + R} \tag{16}$$

The parameter α (α $>=$ 0) provides the ability to assign greater or lesser weight to accuracy. If $\alpha = 1$, recall and accuracy have equal weight, and the F-Measure can be represented using the E score, as expressed by the following formula [39]:

$$F = \frac{2PR}{P+R} = 1 - E \tag{17}$$

C. N+

The N+ metric measures the coverage of vocabulary by the annotation system. It evaluates how many words were correctly assigned to at least one test image, which means it calculates the number of words with positive recall that were successfully assigned by the method [39]. By determining the number of words with positive recall, this metric provides an insight into the amount of vocabulary used by the annotation system.

XII. EVALUATION CRITERIA SELECTED

The performance of the improved automatic image annotation (AIA) system was evaluated by using standard datasets and commonly used measures in the field of image annotation. These measures included recall, accuracy, F-measure, and N+. Recall measures the ability of the system to retrieve all relevant keywords for an image, accuracy measures the overall accuracy of the system in assigning keywords to images, F-measure is the harmonic mean of precision and recall, and N+ measures the amount of vocabulary covered by the system. The annotation rate is another measure of annotation performance [40].

A. EXPERIMENTS RESULTS

Metrics evaluating the efficiency of the ResNet-SLT transformation in feature extraction were analyzed. As shown in Table 3, these descriptors were able to categorize data from three datasets. When compared to other descriptors, the ResNet50-SLT transform clearly excelled. Achieved classification accuracy was around 98% and 95% on the training

TABLE 3. The classification accuracy of the retrieved images obtained using ResNet50-SLT.

Database	Number of Samples	Features Type	Accuracy (%)	
			Training	Testing
Corel-5K	Training 4500	ResNet-SLT	%98	%91
	Testing 500	PLSA-WORDS [31]	%95	%89
ESP-Game	Training 18 689	ResNet-SLT	%93	%91
	Testing 2 081	GAN [29]	%89	%87
Flickr8k	Training 7000	ResNet-SLT	%98	%95
	Testing 1000	SEM [30]	%96	%90

and testing versions of the Flickr8k dataset, respectively. For the Corel-5K dataset, the PLSA-WORDS approach similarly obtained 95% classification accuracy. When examined on the ESP-Game dataset, the GAN [28] method achieved around 89% of classification accuracy, the SEM [29] method achieved as much as 96% of classification accuracy, and the ResNet50-SLT method achieved as much as 98% and 95% of classification accuracy, respectively. Experiments conducted on the Corel5k, ESP Game, and Flickr8k datasets all yielded extremely promising results.

After evaluating the suggested method's accuracy performance on the Corel5k dataset, as a result, reached the following conclusions. ResNet50-SLT was the most precise method (98%), When compared to more standard approaches, ResNet50-SLT and SEM schemes performed exceptionally well in terms of classification accuracy (Table 1 and 2). When applied to the ESP-Game, the accuracy performance of the developed approaches was quite promising. The ResNet50-SLT method has the highest accuracy (91%), followed by the GAN method (87%). ResNet50-SLT and SEM schemes had the best classification performances on images' features compared to other methods that have been published (Table 3).

Figure 15 shows the training and testing dataset accuracy for the fully-connected NN model. The training set accuracy did not exceed the testing set accuracy, implying that the model could not over fit.

B. EXPERIMENTAL SETUP

TPython was used for development, and the deep learning (DL) frameworks Keras [31] and TensorFlow were put into place to create the suggested AIA system [32]. backend. Python was utilized as a sleeve to encapsulate all actions during the testing and training of the dataset.

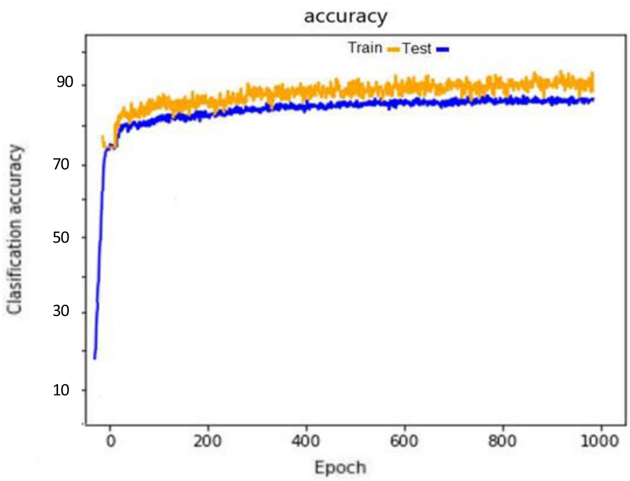
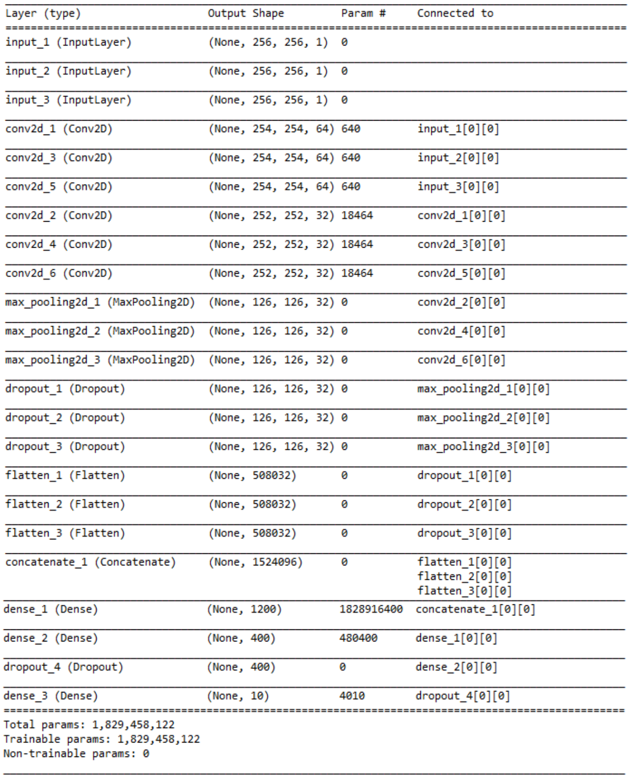


FIGURE 15. Classification accuracy curve for the best fully-connected NN in Flickr8k database.



```
_____________________________________________________________________________________
Layer (type)                   Output Shape         Param #     Connected to
=====================================================================================
input_1 (InputLayer)           (None, 256, 256, 1)  0
_____________________________________________________________________________________
input_2 (InputLayer)           (None, 256, 256, 1)  0
_____________________________________________________________________________________
input_3 (InputLayer)           (None, 256, 256, 1)  0
_____________________________________________________________________________________
conv2d_1 (Conv2D)              (None, 254, 254, 64) 640         input_1[0][0]
_____________________________________________________________________________________
conv2d_3 (Conv2D)              (None, 254, 254, 64) 640         input_2[0][0]
_____________________________________________________________________________________
conv2d_5 (Conv2D)              (None, 254, 254, 64) 640         input_3[0][0]
_____________________________________________________________________________________
conv2d_2 (Conv2D)              (None, 252, 252, 32) 18464       conv2d_1[0][0]
_____________________________________________________________________________________
conv2d_4 (Conv2D)              (None, 252, 252, 32) 18464       conv2d_3[0][0]
_____________________________________________________________________________________
conv2d_6 (Conv2D)              (None, 252, 252, 32) 18464       conv2d_5[0][0]
_____________________________________________________________________________________
max_pooling2d_1 (MaxPooling2D) (None, 126, 126, 32) 0           conv2d_2[0][0]
_____________________________________________________________________________________
max_pooling2d_2 (MaxPooling2D) (None, 126, 126, 32) 0           conv2d_4[0][0]
_____________________________________________________________________________________
max_pooling2d_3 (MaxPooling2D) (None, 126, 126, 32) 0           conv2d_6[0][0]
_____________________________________________________________________________________
dropout_1 (Dropout)            (None, 126, 126, 32) 0           max_pooling2d_1[0][0]
_____________________________________________________________________________________
dropout_2 (Dropout)            (None, 126, 126, 32) 0           max_pooling2d_2[0][0]
_____________________________________________________________________________________
dropout_3 (Dropout)            (None, 126, 126, 32) 0           max_pooling2d_3[0][0]
_____________________________________________________________________________________
flatten_1 (Flatten)            (None, 508032)       0           dropout_1[0][0]
_____________________________________________________________________________________
flatten_2 (Flatten)            (None, 508032)       0           dropout_2[0][0]
_____________________________________________________________________________________
flatten_3 (Flatten)            (None, 508032)       0           dropout_3[0][0]
_____________________________________________________________________________________
concatenate_1 (Concatenate)    (None, 1524096)      0           flatten_1[0][0]
                                                                flatten_2[0][0]
                                                                flatten_3[0][0]
_____________________________________________________________________________________
dense_1 (Dense)                (None, 1200)         1828916400  concatenate_1[0][0]
_____________________________________________________________________________________
dense_2 (Dense)                (None, 400)          480400      dense_1[0][0]
_____________________________________________________________________________________
dropout_4 (Dropout)            (None, 400)          0           dense_2[0][0]
_____________________________________________________________________________________
dense_3 (Dense)                (None, 10)           4010        dropout_4[0][0]
=====================================================================================
Total params: 1,829,458,122
Trainable params: 1,829,458,122
Non-trainable params: 0
_____________________________________________________________________________________
```

FIGURE 16. Summary of the proposed ResNet50-LSTM configuration.

To begin, Python 3.5 was added to the predetermined environment. Python packages TensorFlow and PrettyTensor were then installed. Figure 16 summarizes the proposed ResNet50-LSTM configuration.

When building NNs in TensorFlow, PrettyTensor [41] made the process considerably more streamlined, allowing developers to spend more time on the design model and fewer hours on the implementation details. Using three industry-standard benchmark datasets, the ResNet50-LSTM model's accuracy and consistency were evaluated. To address these fresh AIA issues, researchers turned to the full CNN annotation system. The system integrated and blended the retrieved characteristics using the CNN architecture to maximize the benefits of the unique AIA framework. At first, it defined the datasets and assessed the indicators. Second, the validation results were briefly given and discussed for each methodology. Finally, the performance of the suggested model was evaluated in contrast to existing cutting-edge image annotation techniques. Several examples were also supplied to help illustrate the image annotation process and its underlying principles.

C. RESULTS

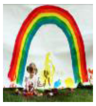
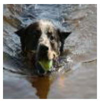
When deep models like ResNet50 are used to annotate images, the overall efficiency of the model increases. When looking at the BLEU metric in particular, this becomes immediately apparent. In this section, we will first examine the function of "soft attention" in AIA.

A significant improvement in the model's performance is shown after the incorporation of the soft attention mechanism. The BLEU efficiency index rises as a result of using the soft attention technique. In addition, when the generator model has been trained, there are two queries to be run. One is whether or not the model actually generates new descriptions, and the other is whether or not those descriptions are diverse, high-quality, and easy to understand for humans. Additionally, we have conducted an additional set of experiments that include human judgment in our overall performance evaluation.

According to the findings, the model with soft attention produces 71% more accurate captions than the one without, and this improvement is sustained throughout a broader range of caption lengths and complexity. As a result, we paid special attention to the soft focus while conducting our AIA-based studies. These are examples of subtitles produced by our proposed ResNet50-SLT system.

Annotation examples (Actual and prediction) from training and testing iterations of each dataset are shown in Table 4. The proposed AIA approach effectively extended the labels The proposed model analyze the results in relation to those of other currently available image adapting infrastructures. The experimental results shown in Table 5 obtained using the public dataset Flickr8k ResNet50-SLT were %85 precision, %83 recall, and %85 f1-score, with a N+ of 285, and for Corel5K database were %78 precision, %80 recall, and %81 f1-score, with a N+ of 290, Esp-Game reached to %84 precision, %78 recall, and %80 f1-score, with a N+ of 270. The proposed model outperforms the alternatives, demonstrating it is good at encapsulating visual information. Our research proves that AIA works well by merging ResNet50-SLT, LSTM, and soft attention into a single model.

TABLE 4. Examples of each dataset used by the proposed AIA system.

Datasets	Prediction	Image Datasets		
Flicker8k	Actual	black dog and spotted dog are fighting	brown dog is running through brown field	man skateboarding
	Prediction	two dogs are playing with each other on the sidewalk	brown dog runs through field of yellow flowers	skateboarder is being splashed by window
				
ESP-GAME	Actual	boxer in black trunks taking swing at boxer in white trunks	little girl covered in paint sits in front of painted rainbow with her hands in bowl	man in hat is displaying pictures next to skier in blue ha
	Prediction	two boxers fight for the ball	the little girl in pigtails is in the rainbow	two children are riding on skis on snowy day
				
Corel5K	Actual	boy is wakeboarding on lake with one hand	blond-hair girl is eating peach	black dog is Swimming while carrying tennis
	Prediction	boy rides wakeboard in the water	little girl eating peach	dog with its mouth open in its mouth swims

D. COMPARATIVE PERFORMANCE EVALUATION OF THE PROPOSED METHODS WITH OTHERS

Table 6 presents the experimental results obtained using the proposed AIA system on three datasets when compared with other works. As mentioned above, the proposed AIA system was implemented in a public DL software Keras [31] and Tensorflow [32]. The weights in the neural networks (NNs) were initialized using Keras, and all layers in the deep network were initialized using the ADADELTA method [33]. The entire network was trained using a Dell Precision T1700 CPU system with 16 GB of memory. The classification

TABLE 5. The results generated by the suggested model.

Name	Database	Precision	Recall	F1-score	N+ (Number of Words)
ResNet50-SLT	Flickr8k	85%	83%	85%	285
	Corel5K	78%	80%	81%	290
	ESP-GAME	84%	78%	80%	270

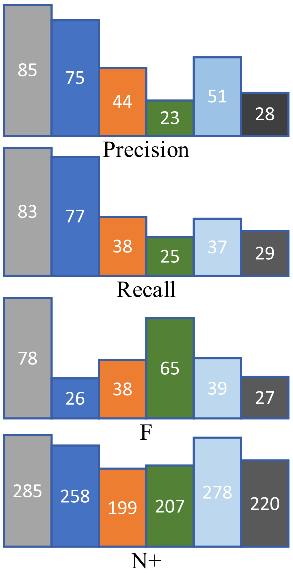


FIGURE 17. A comparison of the results obtained using the proposed approach applied on dataset (Flickr8k) with others.

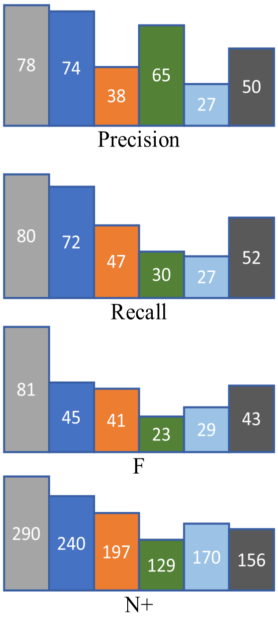


FIGURE 18. A comparison of the results obtained using the proposed approach applied on dataset (Corel5K) with others.

accuracy of the deep learning (DL) system was evaluated using the procedure outlined in Table 6, which summarizes the proposed ResNet50 configuration based on the Kears library. The average experimental values of precision, recall, and F-measure for the proposed ResNet50 model on each dataset were compared to other results found in the literature, as depicted in figure 17.

The proposed AIA-ResNet50-SLT method achieved better results compared to other methods such as CNN-THOP, SEM, and GAN, which are considered more suitable for annotation tasks as they have improved recall and precision. Additionally, when applied to the Flickr8k, Corel5K, and ESP-Game datasets [42], the proposed AIA system achieved higher recall and F-Measure values. Implemented on three different datasets, the newly developed AIA system produced the most effective features through automatic feature extraction and object learning representation. In addition, the proposed AIA system attained the highest F-value compared to other, indicating its effectiveness and robustness. The performance of the proposed AIA approach when implemented on the Flickr8k dataset followed the trend of and shown in figure 17: ResNet50-SLT approach and CNN-THOP attained the highest P (85%). ResNet50-SLT approach achieved the highest R (83%) compared to CNN-THOP (78%); ResNet50-SLT technique produced the highest F1 (85%) compared to all other existing algorithms (66%) with a difference of 19%. Furthermore, the ResNet50-SLT method produced the highest N+ with CNN-THOP (285) compared to the one attained by other algorithms (270) [2], [5], which was improved by at least 3. Results in Table 6.3 clearly indicated that the ResNet50-SLT approach and CNN-THOP provided the largest N+. In addition, the difference in F1 between ResNet50-SLT approach and SEM was discerned to be %21, indicating that ResNet50-SLT approach outperformed the SEM in terms of annotation performance.

The performance of the proposed AIA approach was evaluated using the Corel-5k dataset, as shown in figure 18. It was found that the ResNet50-SLT approach and the CNN-THOP approach achieved the highest precision at (78%). The ResNet50-SLT approach also had the highest recall at (80%), whereas CNN-THOP had (72%). Additionally, the ResNet50-SLT method had the highest F1 score at 81%. Furthermore, the ResNet50-SLT method produced the highest N+ with CNN-THOP (240) compared to the one attained by other algorithms (290) [43], [44], which was improved by at least 4. Additionally, The F1 scores of ResNet50-SLT show a 15% difference between its performance and that of CNN-THOP, indicating that ResNet50-SLT performed better.

The proposed AIA approach was applied to a dataset called ESP-Game performance, as shown in figure 19, and it was found that the ResNet50-SLT approach and the CNN-THOP approach achieved the highest precision at (80%). The CNN-THOP approach also had the highest recall at 80%, whereas the ResNet50-SLT approach had the highest

TABLE 6. Experimental results obtained using the proposed AIA system on three datasets when compared with other works.

Dataset	Corel-5k				ESP-Game				Flickr8k			
Measure	P	R	F	N+	P	R	F	N+	P	R	F	N+
SEM [29]	%50	%52	%43	156	%38	%42	%40	258	%41	%39	%40	284
JEC [35]	%27	%27	%29	170	%22	%25	%23	199	%28	%29	%28	220
PLSA-WORDS [30]	%65	%30	%23	129	%20	%24	%21	201	%23	%25	%23	207
GAN [28]	%38	%47	%41	197	-	-	-	-	%44	%38	%43	199
CNN-THOP [34]	74%	%72	%45	240	%77	%80	%37	250	%80	%78	%66	270
Proposed Model	78%	80%	81%	290	80%	78%	84%	270	85%	83%	85%	285

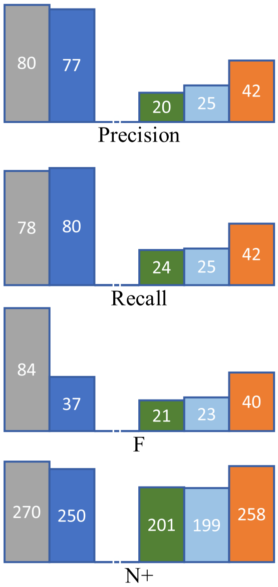


FIGURE 19. A comparison of the results obtained using the proposed approach applied on dataset (ESP-Game) with others.

F1 score at (84%). Additionally, the ResNet50-SLT method had the highest annotation performance with a score of 258, which was 4 points better than the scores of other algorithms. ResNet50-SLT score difference between ResNet50 and SLT and the CNN-THOP approach was (24%), indicating that the ResNet50-SLT approach performed better in annotation.

XIII. CONCLUSION

This paper presents a novel approach for annotating images that combines the features of ResNet50-SLT and information from neighboring images. The system utilizes low-level and high-level features, such as shape, texture, and color, to characterize each image. An algorithm for semantic extension is introduced and its implementation details are provided. This study aimed to address the challenge of accurate image description and retrieval, while also reducing the computational complexity of existing approaches. The proposed AIA system, which combines ResNet50-Slantlet transform and word2vec with principal component analysis and t-distributed stochastic neighbor embedding, achieved impressive results on popular datasets such as Flickr8k, Corel-5k, and ESP-Game. Specifically, the system demonstrated a higher level of accuracy in image retrieval, while also decreasing computing complexity. While this study represents a significant step forward in the field of computer vision and image annotation, further research is needed to address the remaining challenges, such as the semantic gap between low-level computer features and human interpretation of images. The study assesses the impact of various ResNet50 architectures on the experiment results using three datasets: Corel5K, ESP-Game, and Flickr8k. The results indicate that the proposed method balances precision, recall, and F-measure using ResNet50-SLT. While the training phase of an image annotation model can be computationally demanding, especially with large datasets, the proposed method is efficient regarding its results. Compared to traditional methods, the present approach can overcome some challenges of requiring more resources and time to train the model. Despite being computationally intensive, the method is still time-efficient when used with large training datasets. The AIA system suggested in the study underwent a performance evaluation on the Flickr8k dataset, resulting in the highest precision score of 85% and the highest recall score of 83%.

On the other hand, the ResNet50-SLT method scored the highest F1 score of 85%. In addition, the ResNet50-SLT approach generated the highest N+ score with 285 words. The proposed image representation was superior to existing approaches published in the literature after a thorough performance evaluation utilizing experiments, making it a safe bet to be recommended as a preferred way for image retrieval tasks. In conclusion, the visual words integration of ResNet50-SLT can provide good retrieval performance, in addition to the advantages of fast indexing and scalability, depending on the image collection.

ACKNOWLEDGMENT

This research was funded by Princess Nourah bint Abdulrahman University and Researchers Supporting Project number (PNURSP2023R346), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia. The authors would also like to thank Prince Sultan University, Riyadh KSA for their support.

REFERENCES

[1] M. S. Rahim, M. Norouzi, and A. Rehman, “3D bones segmentation based on CT images visualization,” *Biomed. Res.*, vol. 28, no. 8, pp. 3641–3644, 2017.

[2] J. Tang, X. Shu, G.-J. Qi, Z. Li, M. Wang, S. Yan, and R. Jain, “Tri-clustered tensor completion for social-aware image tag refinement,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 8, pp. 1662–1674, Aug. 2017.

[3] T. Saba, A. Rehman, and G. Sulong, “An intelligent approach to image denoising,” *J. Theor. Appl. Inf. Technol.*, vol. 17, no. 2, pp. 32–36, 2010.

[4] Q. Cheng, Q. Zhang, P. Fu, C. Tu, and S. Li, "A survey and analysis on automatic image annotation," *Pattern Recognit.*, vol. 79, pp. 242–259, Jul. 2018.

[5] J. Tang, X. Shu, Z. Li, Y.-G. Jiang, and Q. Tian, "Social anchor-unit graph regularized tensor completion for large-scale image retagging," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 41, no. 8, pp. 2027–2034, Aug. 2019.

[6] T. Gui, R. Ma, Q. Zhang, L. Zhao, Y.-G. Jiang, and X. Huang, "CNN-based Chinese NER with lexicon rethinking," in *Proc. 28th Int. Joint Conf. Artif. Intell.*, Aug. 2019, pp. 4982–4988.

[7] N. Abbas, T. Saba, D. Mohamad, A. Rehman, A. S. Almazyad, and J. S. Al-Ghamdi, "Machine aided malaria parasitemia detection in Giemsa-stained thin blood smears," *Neural Comput. Appl.*, vol. 29, no. 3, pp. 803–818, Feb. 2018.

[8] U. Sharif, Z. Mehmood, T. Mahmood, M. A. Javid, A. Rehman, and T. Saba, "Scene analysis and search using local features and support vector machine for effective content-based image retrieval," *Artif. Intell. Rev.*, vol. 52, no. 2, pp. 901–925, Aug. 2019.

[9] T. Saba, S. T. F. Bokhari, M. Sharif, M. Yasmin, and M. Raza, "Fundus image classification methods for the detection of glaucoma: A review," *Microsc. Res. Technique*, vol. 81, no. 10, pp. 1105–1121, Oct. 2018.

[10] M. Mundher, D. Muhamad, A. Rehman, T. Saba, and F. Kausar, "Digital watermarking for images security using discrete slantlet transform," *Appl. Math. Inf. Sci.*, vol. 8, no. 6, pp. 2823–2830, Nov. 2014.

[11] M. Yousuf, Z. Mehmood, H. A. Habib, T. Mahmood, T. Saba, A. Rehman, and M. Rashid, "A novel technique based on visual words fusion analysis of sparse features for effective content-based image retrieval," *Math. Problems Eng.*, vol. 2018, pp. 1–13, Jan. 2018.

[12] V. Kovalev and M. Petrou, "Multidimensional co-occurrence matrices for object recognition and matching," *Graph. Models Image Process.*, vol. 58, no. 3, pp. 187–197, May 1996.

[13] X.-Y. Wang, B.-B. Zhang, and H.-Y. Yang, "Content-based image retrieval by integrating color and texture features," *Multimedia Tools Appl.*, vol. 68, no. 3, pp. 545–569, Feb. 2014.

[14] J. W. Z. Zhang, "Content-based image retrieval using color and edge direction features," in *Proc. 2nd Int. Conf. Adv. Comput. Control*, Mar. 2010, pp. 459–462.

[15] T. Saba, A. Rehman, Z. Mehmood, H. Kolivand, and M. Sharif, "Image enhancement and segmentation techniques for detection of knee joint diseases: A survey," *Current Med. Imag. Rev.*, vol. 14, no. 5, pp. 704–715, Sep. 2018.

[16] A. Sarwar, Z. Mehmood, T. Saba, K. A. Qazi, A. Adnan, and H. Jamal, "A novel method for content-based image retrieval to improve the effectiveness of the bag-of-words model using a support vector machine," *J. Inf. Sci.*, vol. 45, no. 1, pp. 117–135, Feb. 2019.

[17] G. Zhang, C.-H.-R. Hsu, H. Lai, and X. Zheng, "Deep learning based feature representation for automated skin histopathological image annotation," *Multimedia Tools Appl.*, vol. 77, no. 8, pp. 9849–9869, Apr. 2018.

[18] V. N. Murthy, A. Sharma, V. Chari, and R. Manmatha, "Image annotation using multi-scale hypergraph heat diffusion framework," in *Proc. ACM Int. Conf. Multimedia Retr.*, Jun. 2016, pp. 299–303, doi: 10.1145/2911996.2912055.

[19] M. Guillaumin, T. Mensink, J. Verbeek, and C. Schmid, "TagProp: Discriminative metric learning in nearest neighbor models for image auto-annotation," in *Proc. IEEE 12th Int. Conf. Comput. Vis.*, Sep. 2010, pp. 309–316.

[20] A. Rehman and T. Saba, "Off-line cursive script recognition: Current advances, comparisons and remaining problems," *Artif. Intell. Rev.*, vol. 37, no. 4, pp. 261–288, Apr. 2012.

[21] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[22] Y. Verma and C. V. Jawahar, "Image annotation using metric learning in semantic neighbourhoods," in *Proc. Eur. Conf. Comput. Vis.*, vol. 7574, 2012, pp. 836–849, doi: 10.1007/978-3-642-33712-3_60.

[23] Q. Liao, L. Jiang, X. Wang, C. Zhang, and Y. Ding, "Cancer classification with multi-task deep learning," in *Proc. Int. Conf. Secur., Pattern Anal., Cybern. (SPAC)*, Dec. 2017, pp. 76–81, doi: 10.1109/SPAC.2017.8304254.

[24] F. M. Rammo and N. M. Al-Hamdani, "Detecting the speaker language using CNN deep learning algorithm," *Iraqi J. For Comput. Sci. Math.*, vol. 3, no. 1, pp. 43–51, 2022.

[25] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*.

[26] M. S. Haji, M. H. Alkawaz, and A. Rehman, "Content-based image retrieval: A deep look at features prospectus," *Int. J. Comput. Vis. Robot.*, vol. 9, no. 1, pp. 14–38, 2019.

[27] M. M. Adnan, M. S. M. Rahim, A. R. Khan, T. Saba, S. M. Fati, and S. A. Bahaj, "An improved automatic image annotation approach using convolutional neural network-slantlet transform," *IEEE Access*, vol. 10, pp. 7520–7532, 2022.

[28] F. Gao, S. Ji, J. Guo, Q. Li, Y. Ji, Y. Liu, S. Feng, H. Wei, N. Wang, and B. Yang, "ID-Net: An improved mask R-CNN model for intrusion detection under power grid surveillance," *Neural Comput. Appl.*, vol. 33, no. 15, pp. 9241–9257, Aug. 2021.

[29] Y. Ma, Y. Liu, Q. Xie, and L. Li, "CNN-feature based automatic image annotation method," *Multimedia Tools Appl.*, vol. 78, no. 3, pp. 3767–3780, Feb. 2019.

[30] T. Uricchio, L. Ballan, L. Seidenari, and A. Del Bimbo, "Automatic image annotation via label transfer in the semantic space," *Pattern Recognit.*, vol. 71, pp. 144–157, Nov. 2017.

[31] F. Chollet. (2015). *Keras Documentation*. Keras.Io. [Online]. Available: https://keras.io

[32] Z. Mehmood, M. Rashid, A. Rehman, T. Saba, H. Dawood, and H. Dawood, "Effect of complementary visual words versus complementary features on clustering for effective content-based image search," *J. Intell. Fuzzy Syst.*, vol. 35, no. 5, pp. 5421–5434, Nov. 2018.

[33] M. D. Zeiler, "ADADELTA: An adaptive learning rate method," 2012, *arXiv:1212.5701*.

[34] J. Cao, A. Zhao, and Z. Zhang, "Automatic image annotation method based on a convolutional neural network with threshold optimization," *PLoS ONE*, vol. 15, no. 9, Sep. 2020, Art. no. e0238956.

[35] A. Makadia, V. Pavlovic, and S. Kumar, "A new baseline for image annotation," *Int. J. Comput. Vis.*, vol. 90, pp. 88–105, Jan. 2010.

[36] H. Kwasnicka and M. Paradowski, "On evaluation of image auto-annotation methods," in *Proc. 6th Int. Conf. Intell. Syst. Design Appl.*, Oct. 2006, pp. 353–358, doi: 10.1109/ISDA.2006.253861.

[37] X. Liu, L. Faes, A. U. Kale, S. K. Wagner, D. J. Fu, A. Bruynseels, T. Mahendiran, G. Moraes, M. Shamdas, C. Kern, J. R. Ledsam, M. K. Schmid, K. Balaskas, E. J. Topol, L. M. Bachmann, P. A. Keane, and A. K. Denniston, "A comparison of deep learning performance against health-care professionals in detecting diseases from medical imaging: A systematic review and meta-analysis," *Lancet Digit. Health*, vol. 1, no. 6, pp. e271–e297, Oct. 2019.

[38] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," 2013, *arXiv:1301.3781*.

[39] I. W. Selesnick, "The slantlet transform," *IEEE Trans. Signal Process.*, vol. 47, no. 5, pp. 1304–1313, May 1999.

[40] T. Saba, A. Rehman, A. Altameem, and M. Uddin, "Annotated comparisons of proposed preprocessing techniques for script recognition," *Neural Comput. Appl.*, vol. 25, no. 6, pp. 1337–1347, Nov. 2014.

[41] L. von Ahn and L. Dabbish, "Labeling images with a computer game," in *Proc. SIGCHI Conf. Hum. Factors Comput. Syst.*, Apr. 2004, pp. 1–4.

[42] H. Younis, M. H. Bhatti, and M. Azeem, "Classification of skin cancer dermoscopy images using transfer learning," in *Proc. 15th Int. Conf. Emerg. Technol. (ICET)*, 2019, pp. 1–4.

[43] K. T. Ahmed, S. A. H. Naqvi, A. Rehman, and T. Saba, "Convolution, approximation and spatial information based object and color signatures for content based image retrieval," in *Proc. Int. Conf. Comput. Inf. Sci. (ICCIS)*, Apr. 2019, pp. 1–6.

[44] A. Khan, "Improved multi-lingual sentiment analysis and recognition using deep learning," *J. Inf. Sci.*, vol. 76, 2023, Art. no. 01655515221137270.

MYASAR MUNDHER ADNAN received the B.E. degree in computer science from Alkufa University, Iraq, in 2011, and the M.S. degree in computer science from UTM, in 2014, where he is currently pursuing the Ph.D. degree. His current research interests include deep learning, image processing, and machine learning.

MOHD SHAFRY MOHD RAHIM received the B.Sc. (Hons.) and M.Sc. degrees in computer science from Universiti Teknologi Malaysia, in 1999 and 2002, respectively, and the Ph.D. degree in spatial modeling from Universiti Putra Malaysia, in 2008. He is currently a Professor in image processing with the School of Computing, Universiti Teknologi Malaysia. He has appointed to be the Deputy Director of the Centre for Joint Program, UTMSPACE. He also focused his research together with his research group, with the UTM ViCube Laboratory, Faculty of Computing, UTM. He is also an expert in the research area of computer graphics and image processing.

AMJAD REHMAN KHAN (Senior Member, IEEE) received the Ph.D. degree from the Faculty of Computing, Universiti Teknologi Malaysia, with a specialization in forensic documents analysis and security, in 2010. He is currently a Senior Researcher with the Artificial Intelligence and Data Analytics Laboratory, Prince Sultan University, Riyadh, Saudi Arabia. He is also a PI in several funded projects and also completed projects funded from MOHE Malaysia, Saudi Arabia. He is the author of more than 200 ISI journal articles and conferences. His H-index is 40 with 4000 citations. His research interests include data mining, health informatics, and pattern recognition. He received a Rector Award for the 2010 Best Student from Prince Sultan University.

AHMED ALKHAYYAT received the B.Sc. degree in electrical engineering from Al Kufa University, Najaf, Iraq, in 2007, and the M.Sc. degree from the Dehradun Institute of Technology, Dehradun, India, in 2010. He is currently the Dean of International Relationship and the Manager of the word ranking with Islamic University, Najaf. His research interests include network coding, cognitive radio, efficient-energy routing algorithms, efficient-energy MAC protocol in cooperative wireless networks and wireless local area networks, and cross-layer designing for self-organized networks. He contributed in organizing several IEEE conferences, workshop, and special sessions. To serve his community, he acted as a reviewer for several journals and conferences.

FATEN S. ALAMRI received the Ph.D. degree in system modeling and analysis in statistics from Virginia Commonwealth University, USA, in 2020. Her Ph.D. research was in Bayesian dose response modeling, experimental design, and nonparametric modeling. She is currently an Assistant Professor with the Department of Mathematical Sciences, College of Science, Princess Nourah Bint Abdul Rahman University. Her research interests include spatial area, environmental statistics, and brain imaging.

TANZILA SABA (Senior Member, IEEE) received the Ph.D. degree in document information security and management from the Faculty of Computing, Universiti Teknologi Malaysia (UTM), Malaysia, in 2012. She is currently an Associate Professor with the College of Computer and Information Sciences, Prince Sultan University (PSU), Riyadh, Saudi Arabia. She has published more than 100 publications in high ranked journals. Her primary research interests include bioinformatics, data mining, and classification. She won the Best Student Award with the Faculty of Computing, UTM, in 2012. She was awarded the Best Research of the Year Award in PSU, from 2013 to 2016. Due to her excellent research achievement, she is included in Marquis Who's Who (S & T) 2012. She is also an editor of several reputed journals and on panel of TPC of international conferences.

SAEED ALI BAHAJ received the Ph.D. degree from Pune University, India, in 2006. He is currently an Associate Professor with the Computer Engineering Department, Hadramout University, Yemen, and MIS Department, COBA, Prince Sattam Bin Abdul-Aziz. His research interests include artificial intelligence, information management, forecasting, information engineering, big data, and information security.