IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 12 June 2023, accepted 18 July 2023, date of publication 24 July 2023, date of current version 31 July 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3298216

RESEARCH ARTICLE

Stacked Siamese Neural Network (SSiNN) on Neural Codes for Content-Based Image Retrieval

GOPU V. R. MUNI KUMAR AND D. MADHAVI
Department of ECE, GITAM School of Technology, GITAM Deemed to be University, Rushikonda, Visakhapatnam 530045, India
Corresponding author: Gopu V. R. Muni Kumar (mgopu@gitam.in)

ABSTRACT Content-based image retrieval (CBIR) represents a class of problems that aims at finding relevant images in response to an image-based search query. The CBIR systems use similarity measures or distance metrics between a group of representative features in the query image and those in the image repository. Traditionally, these features were generated by hand, employing image features such as colour, texture, shape, and so on. Due to the fact that these methods do not provide a comprehensive perspective of the images, they cannot be widely utilized in contemporary CBIR systems. This is due to the so-called semantic gap between query intent and system perspective. The most recent advancements in deep learning offer a viable alternative to manually built features, leveraging the representational learning capability of deep neural networks. This paper presents a method of implementing a CBIR system using a multi-stage approach known as classify, differentiate, and retrieve (CDR). The first stage involves using a deep neural network to encode the images. Later, a custom-trained stacked Siamese Neural network (SSiNN) is employed to differentiate the latent space representation of the images obtained from the first stage. The experimental results for the CIFAR-10 dataset were presented, along with an algorithm for applying this strategy to any generic dataset. Experimental outcomes demonstrate that the proposed strategy is superior to the current best practices.

INDEX TERMS Content-based image retrieval, CBIR, deep learning, semantic gap, siamese network.

I. INTRODUCTION

Image retrieval systems have been extensively researched, with approaches ranging from handcrafted features [22], [23], [24] to the most recent deep learning-based solutions [25], [26]. The term ''content-based image retrieval'' is used to refer to a group of problems involving the retrieval of relevant images from a repository of images in response to image-based queries. One of the vital components of the CBIR system is a similarity metric [27]. An optimal similarity metric has a low value for relevant Images and a high value for irrelevant images.

Definition 1: For a given set X, for any $x, y, z \in X$*, a real-valued function* $\lambda(x, y)$ *defined on the cartesian Product* $X \times X$ *is a similarity metric [27] if it satisfies the following conditions:*

- $\lambda(x, x) = 0$
- $\lambda(x, y) = \lambda(y, x)$
- $\lambda(x, z) < \lambda(x, y)$*, if x is similar to z but not to y*
- $\lambda(x, y) = \lambda(x, z) + M$*, where M is the margin, and the objective of learning system is to maximize M*

The simplest of these measures is the template-matching method, which involves calculating the Euclidean or Manhattan distance between the pixel values of the source and target images [1]. One drawback of this method is that each image may have different lighting, orientation, size, dimensions, background clutter, direction of capture, etc. This can result in the unsuccessful retrieval of similar images with such differences, which is counterintuitive.Â Furthermore, this technique for retrieving images requires a lot of processing power. Approaches with computationally intensive retrieval methodologies are not viable for modern CBIR systems as the size of the databases is growing exponentially because of the increase and ease of image capture devices.

To get around this, researchers shifted their focus to concentrating on image retrieval based on their content.

The associate editor coordinating the review of this manuscript and approving it for publication was Yizhang Jiang.

Fundamental techniques in this class involve simple comparisons of the query image's extracted features to those of database images. The features are constructed using properties of images, such as colour, texture, shape, and spatial data [2]. Using statistics, histograms, etc. [3] based on the pixel values of the images can help improve the methods even further. Image feature extraction methods like SIFT, Binarized statistical image features, edge descriptors, etc., are employed in more advanced approaches. Approaches that use Gabor filters and genetic algorithms were illustrated in [7], [8], and [9]. The effectiveness of the feature representations and their discriminatory power determine the overall performance of the image retrieval.

With the latest developments in Deep Learning and the representation learning capability of artificial neural networks, the method of automatic feature generation for CBIR systems has been investigated. The architecture of the CNNs consists of multiple layers of filters and other mathematical transformations that produce various feature representations of the input images at various levels of abstraction. These developments have given a huge opportunity to leverage the capabilities of CNNs and Deep Learning for CBIR tasks [4].

Deep learning utilizes neural networks comprised of multiple layers, including fully connected,convolutional, Pooling, and flattening layers. One advantage of using deep learning or Deep Neural networks for tasks such as pattern recognition or computer vision is feature representation. Depending on the input data, the model can be trained and optimized to cater to the needs of the task at hand. Another advantage is that the approach used is kind of domain-independent. Hence, a computer vision task meant for remote sensing, another for public security, and a third one for healthcare follows the same design principles.

Deep learning is effective when an enormous amount of training data is available for neural network model training.

With the increased use of data capture devices like digital cameras, mobile phones, CCTVs, data generated from streaming platforms like YouTube and social media platforms like Facebook, Instagram, and LinkedIn, as well as tools that enable the sharing of the generated data, a large repository of unstructured data has been generated.

Limitations observed in prior methodologies:

We summarize a few of the drawbacks of existing methodologies that inspired us to work on SSiNNs.

1) Semantic-Gap Issue: Although deep learning-based content-based image retrieval (CBIR) systems surpass non-deep learning approaches, it remains uncertain whether this improvement is attributed to the reduction in semantic gap or merely a result of mathematical approximation accomplished by the deep learning model.
2) Inter-class relationship: Some of the existing approaches employ classification models, either through pre-trained models or custom-built models. These methods use cross-entropy as their loss function, which has the disadvantage of not capturing inter-class relationships [15]. As a result, there is room for improvement.
3) High-dimensional operations: The high dimensionality of feature vectors is another trait shared by deep-learning-based techniques. Some of the approaches employ dimensionality reduction techniques such as PCA (Principal Component Analysis) or VAEs (Variational AutoEncoders) [21]. These approaches may mathematically reduce the dimension without impacting the similarity metrics, but they exacerbate the semantic-gap problem.

To overcome the gaps in existing techniques, we present the Stacked Siamese Neural Network, which employs a two-stage strategy to tackle the CBIR problem. The motivation for proposing a two-stage approach is that we want to encode images without being constrained by learning interclass relationships, and when we differentiate, we work with a latent space representation that can learn interclass relationships and thus transform the initial latent space representations. This work attempts to develop a new CBIR methodology that can be used for any dataset by dividing the total problem into two stages, each of which is optimized separately, therefore outperforming existing methodologies.

Our contributions:

1) Introduced a novel content-based image retrieval method known as Stacked Siamese Neural Network (SSiNN).
2) The phases involved in developing an SSiNN for any dataset for a CBIR task, including the retrieval approach, were discussed.
3) For the purpose of evaluating the proposed CBIR approach, a neural network architecture has been developed and presented. This architecture makes use of a pre-trained model and transfer learning for the first stage and a custom model for the second stage.
4) The details of the dataset that was used for the analysis and the metrics that were used for the evaluation have been outlined.
5) The experimental results of the proposed method in comparison to existing methodologies are presented.

II. RELATED WORK

Utilizing Deep Learning for CBIR has been a topic of study for a decade. In this section, we shall describe some of the most prominent strategies utilized by prior researchers. Abdel-Nabi et al. [5] presented a deep learning-based image retrieval method that employs the mathematically integrated output from several AlexNet model layers to represent the corresponding images. Camlica et al. [6] utilized an autoencoder-based method to determine the significance of picture blocks and thus achieve the retrieval task. Shakarami and Tarrah [10] used an approach that systematically combined the output from the fully connected layer towards the output of AlexNet, HOG, and LBP feature vectors, which were further reduced in dimension by using PCA to produce the final image descriptor.

Kruthika et al. [11] in their work for early detection of Alzheimer's disease using images from MRI scans, using an ensemble model consisting of a Capsule Network, Convolution Neural Network, and an Auto Encoder. Capsule networks [12] were developed to address the orientation problem encountered by Convolution Neural Networks. The difficulty with orientation in CNNs is caused by the pooling layers, which result in data compression. Capsule networks solve the challenge by employing dynamic routing methods to assess object characteristics like posture, velocity, and texture.

Using Siamese or triplet networks [13], a variant of deep neural networks, was offered as a solution to the orientation problem encountered by CBIR systems when employing CNNs.

Cai et al. [14] has also created a piece of work that is related to the Siamese networks. In this framework, the network would take two images as input, and it would be trained in such a way that, if the images were similar, they would be encoded with features that tend to be similar through a mechanism that is known as a weight sharing mechanism. If the images were dissimilar, they would be encoded with features that tend to be dissimilar.

The field of remote sensing is one in which there is relatively little available data. CBIR in remote sensing applications using CNNs is not straightforward because training CNNs from scratch necessitates a massive amount of training data. Liu et al. [15] achieved CBIR for remote sensing applications by using Transfer learning and Siamese networks with a weighted Wasserstein ordinal loss function.

Öztürk et al. [16] research on CBIR for medical images employs a stacked autoencoder-based feature representation for the images and an over-sampling strategy to deal with data scarcity.

III. STACKED SIAMESE NEURAL NETWORK

In this section, the methodology used in the paper is explained. Various components that form the building blocks of the overall solution proposed are presented. The overall framework can be divided into two stages.

A. STAGES OF THE FRAMEWORK

The first stage of the solution is known as encoding, and the second stage is known as the differentiating stage.

- A deep neural network is used in the first stage to transform the images into a latent space representation.
- In the second stage, a pair of neural networks are used for further transforming the latent space representations, such that the resulting neural codes from the neural networks are mapped as close as possible for similar images and as far as possible for dissimilar ones. This would help the CBIR system to cleanly differentiate and thus, retrieve relevant images as precisely as possible.

Role of the two stages in Image retrieval:

- The first stage, also known as the encoding stage of the proposed model, focuses on transforming the images into a high-quality feature representation, disregarding any concerns regarding relationships between different classes. While this feature representation is effective for certain tasks, it lacks the ability to capture inter-class relationships.
- In the second phase of the model, known as the differentiation stage, the feature representation is further transformed into a different representation that effectively captures the inter-class relationships.

In the subsequent sections, we present a detailed description of the different steps associated with creating a fully functional Content-Based Image Retrieval (CBIR) system using the suggested approach. Figure 1 visually represent all the stages involved in both phases.

B. ENCODING STAGE

The goal of this stage is to encode the images, both the database images as well as the query images. To achieve this, we would leverage the capabilities of convolutional neural networks, which are known to be very effective at feature representation.

The output of a CNN with convolution, pooling, and activation layers can be expressed as follows:

$$f(I) = pool(\sigma(W \circledast I) + b) \tag{1}$$

here, I represents the CNN input, $pool$ refers to the pooling layer, σ denotes the activation function, and $\circledast$ represents the convolution operation.

The loss function is categorical cross-entropy, and its mathematical equation is as follows:

$$H(p, q) = -\sum_{x} p(x) log[q(x)] \tag{2}$$

here $p(x)$ and $q(x)$ represent the probability distributions of class X in target and prediction, respectively.

Building a custom deep convolutional neural network would be both time-consuming and computationally intensive. Hence, we would leverage the transfer learning framework, where we use a pre-trained model and repurpose it for the current dataset. The pre-trained model used in this paper is the VGG-16 network with ImageNet weights. For this network, toward the end, three fully connected layers of dimensions 1024, 512, and 256 neurons are added and are represented as FC2, FC1, and FCO.

- FC2: Dense layer with 1024 Neurons
- FC1: Dense layer with 512 Neurons
- FCO: Dense layer with 256 Neurons

FCO is the output layer for the first phase network.

The stage 1 network is shown in Figure 2.

C. NETWORK STRUCTURE OF ENCODING STAGE NEURAL NETWORK

Convolution Blocks: The network comprises five sets of convolutional blocks, each containing several convolutional

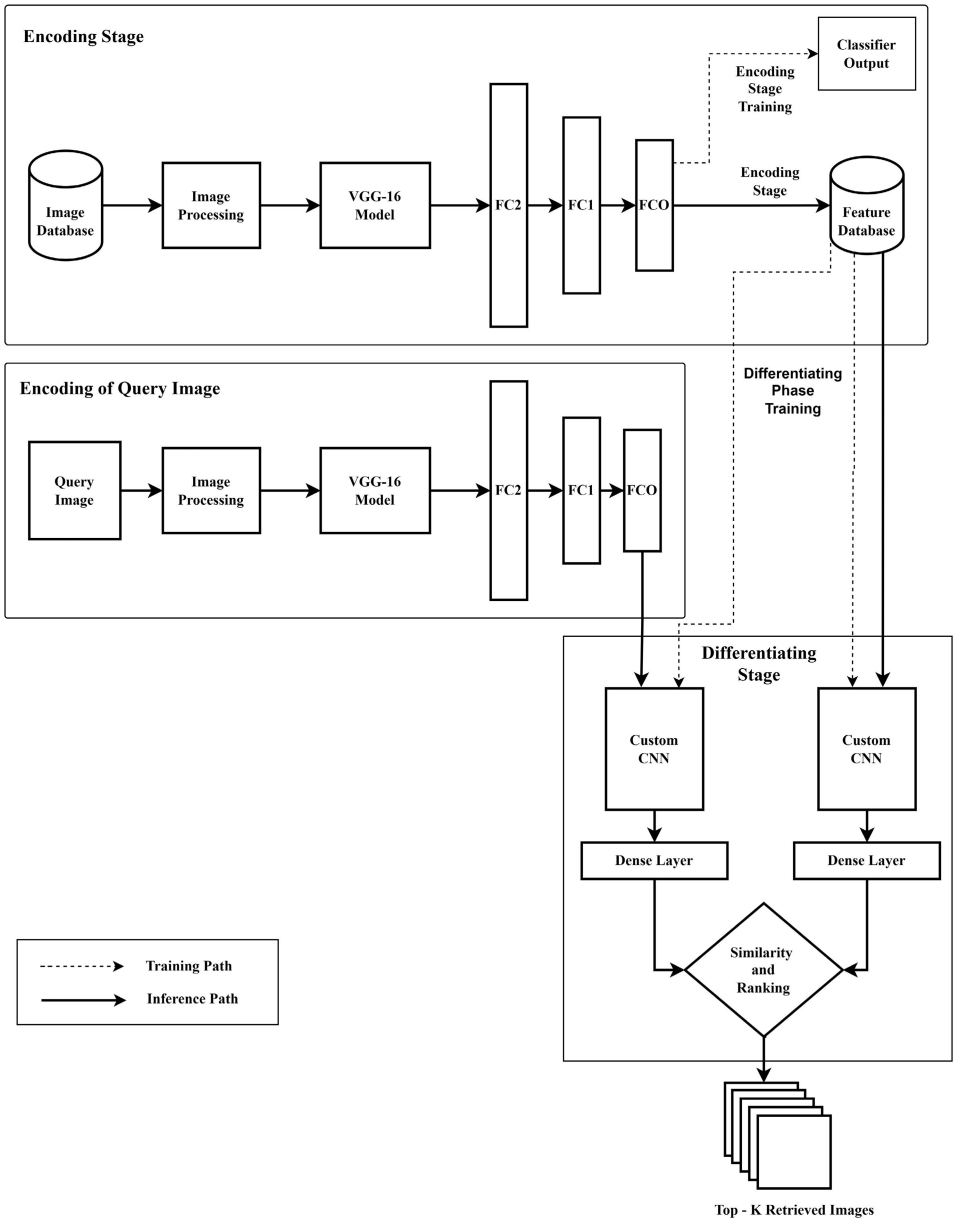

FIGURE 1. The SSINN system operates in two distinct phases. In the encoding phase, images from the image database undergo processing through the initial neural network model to obtain feature representations. It is important to note that certain steps marked by dotted lines are exclusive to the training process. During training, the same database images are utilized to train the neural network. Additionally, when encoding a query image, the same first-stage neural network is employed to acquire the feature representation for the query image. In the differentiating phase, the feature representations obtained during the encoding stage are employed to train the neural network model specific to that stage. This trained model is employed for image retrieval, where one of the sub-networks is fed with the feature representation of the query image, and the other sub-network is fed with the contents of the feature database, which was constructed during the encoding stage.

layers that are then followed by max pooling for down-sampling. The details of each configuration are outlined below.

- Block 1: Two convolutional layers, each employing 64 filters of dimensions 32×32
- Block 2: Two convolutional layers, each incorporating 128 filters of dimensions 16×16
- Block 3: Three convolutional layers, each composed of 256 filters of size 8×8
- Block 4: Three convolutional layers, each equipped with 512 filters of size 4×4

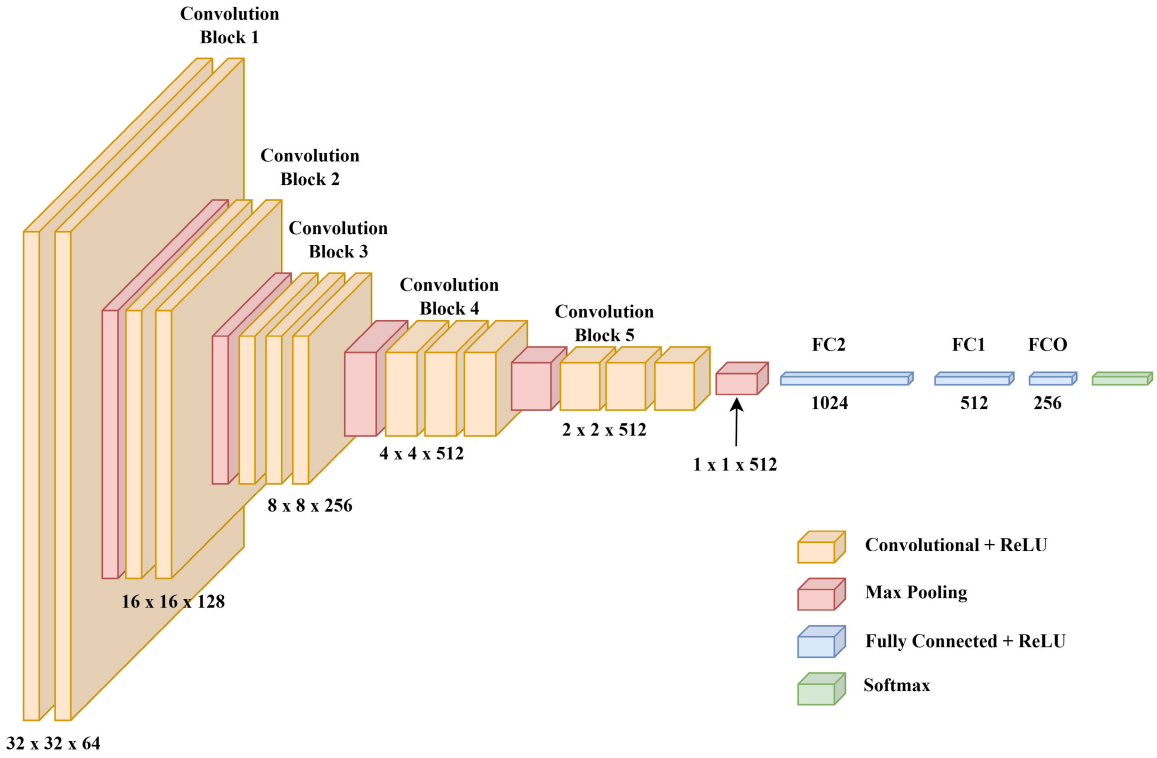

FIGURE 2. The neural network architecture for stage one.

- Block 5: Three convolutional layers, each utilizing 512 filters of dimensions 2×2

Each of the convolution layers uses the 'same' padding and ReLU activation function. After each of the convolution blocks, a 2×2 max pooling with stride 2 is applied.

Fully Connected Layers: Following the convolutional blocks, the network includes three fully connected layers. The first fully connected layer comprises 1024 neurons, followed by the second fully connected layer with 512 neurons. Lastly, the third fully connected layer consists of 256 neurons. The activation function used for all the fully connected layers is ReLU activation.

Output Layer: The output layer is a dense layer with the number of neurons corresponding to the number of target classes; in this case, it is 10. And the activation function for this layer is SoftMax.

D. TRAINING OF ENCODING STAGE

The neural network described above will be trained using the dataset on which the CBIR task has to be performed on the classification task. Once the training is complete, the output of the layer **FCO** will be the latest space representation of the input image.

E. ENCODING OF THE IMAGES

All of the images in the database are encoded using the model that was built during the training process. The input layer of the CNN is the input, and the FCO layer is the output of the CNN model.

F. DIFFERENTIATING STAGE

In the differentiating stage, a network architecture called the Siamese neural network is used. Siamese neural networks are a special type of neural network architecture composed of two identical subnetworks with the same weights and parameters. The reason for utilizing Siamese Neural Networks (SiNN) in a stacked fashion in the current framework is their ability to learn embeddings that position similar classes close enough in the vector space of the embeddings. In the existing research, SiNNs were used directly on the images to design the image retrieval systems. However, in this research, we are using SiNNs in a stacked manner by consuming the latent space representation of the first stage rather than directly working on the images. So far as we know, this is the first time a stacked Siamese neural network (SSiNN) has been used for a CBIR task.

Advantages of using a stacked Siamese neural network (SSiNN):

1) In this approach, we divided the whole framework into two stages: the first stage does the encoding part and uses the cross-entropy measure as its loss function. Cross entropy has the disadvantage of being very good at differentiating classes but not so good at determining the similarity of embeddings. Hence, using the second stage to solely differentiate would complement the shortcomings of the first stage.

2) Using the Siamese network directly on the images, as in previous approaches, increases the number of parameters on which the network must operate. This can lead to overfitting issues, generalization failures, and increased training effort.

The loss function used by the Siamese neural network for the experiments here is Contrastive Loss, which is defined below:

$$CL = (1 - Y) \times \|x_i - x_j\|^2 + Y \times max(0, \alpha - \|x_i - x_j\|^2) \tag{3}$$

here CL stands for Contrastive loss x_i and x_j are the embeddings of the images I_i and I_j respectively from the dataset.
If the overall transformation on the images is Φ, then $x_i = \Phi(I_i)$ and $x_j = \Phi(I_j)$.
Y is label variable, which is defined as:

$$Y = \begin{cases} 0, & \text{if } x_i \text{ is similar to } x_j \\ 1, & \text{otherwise} \end{cases} \tag{4}$$

α is a hyperparameter, which defines lower bound distance for dissimilar samples.

- If the samples are similar ($Y = 0$), then we minimize the term $\|x_i - x_j\|^2$ that corresponds to their Euclidean distance.
- Else, ($Y = 1$), then we minimize the term $max(0, \alpha - \|x_i - x_j\|^2)$ that is equivalent to maximizing their Euclidean distance until some limit α.

The block diagram of SNN is as shown in Figure 3

G. NETWORK STRUCTURE OF DIFFERENTIATING STAGE NEURAL NETWORK

Sub-Networks of Siamese Neural Network: Each of the sub-network comprises three convolutional layers, followed by average pooling for down-sampling. The specific configurations are as follows: The convolution layers in each sub-network are 1D convolutions. The first convolution layer has 4 filters, the second has 16 filters, and the third has 64 filters. The kernel size for each convolution layer is 5, and the activation function used is ReLU. After each convolution layer, there is an Average Pooling layer with a pool size of 2.

The sub-network outputs are merged through a block that computes the Euclidean distance, subsequently applying a batch-normalization layer and a sigmoid activation.

H. TRAINING OF DIFFERENTIATING STAGE NEURAL NETWORKS

During the differentiating stage, the specified network will be trained with the input dataset consisting of the Latent space representations of the images produced during the encoding stage.
Data Preparation for Training: Create Pairs of Latent Space Representations:

The objective of the model is to distinguish embeddings if they correspond to distinct image classes in the dataset. Random embeddings from class A are coupled with random images from class B. Here A and B are some arbitrary distinct classes. The method is repeated for each class. As these pairs belong to distinct classes, the network's output should be 1, as stated by the variable Y in the definition of contrast loss. The network is trained using this labeled data with Y as the output variable.

The network topology and the parameters of the individual subnetworks of the SiNN are shown in Table 1. And the network topology and parameters of the overall SiNN are shown in Table 2. The total number of parameters in the SiNN is 188,086; out of these, the number of trainable parameters is 188,082.

The performance comparison at K=20 is shown in Table 5. In this context, the variable "K" represents the query input used as an argument within the Content-Based Image Retrieval (CBIR) system. It instructs the system to retrieve the top-K similar images from the image database.

I. PREDICTION FROM DIFFERENTIATING STAGE

When the latent space representation of two images is fed to the model constructed during the differentiating stage, the model will predict whether or not the images are similar.

J. RETRIEVAL METHOD

The content-based image retrieval process described in this paper can be summed up by the steps below.

1) Encode all the images from the database and the input query image using the model from the encoding stage to obtain the latent space representations of these images.
2) Use the differentiating network model to find the similarity.
3) One of the inputs to the Siamese network is the latent space representation of the query image, and the other input is the latent space representation of each of the database images.
4) Based on the output of the Siamese network, rank the database images.
5) Furnish the top-k ranked images from the previous step to retrieve the top-k similar images.

K. ALGORITHMIC REPRESENTATION OF THE PROPOSED METHODOLOGY

Training Algorithm:

1) Train the proposed stage-I neural network using the training data to create the encoding model.
2) Employ the encoding model built in Step 1 to perform inference on the training data images, thereby obtaining their latent space representations.
3) Utilize the procedure described in the methodology to create pairs of latent space representations for images. Label these pairs based on the similarity of the corresponding images.

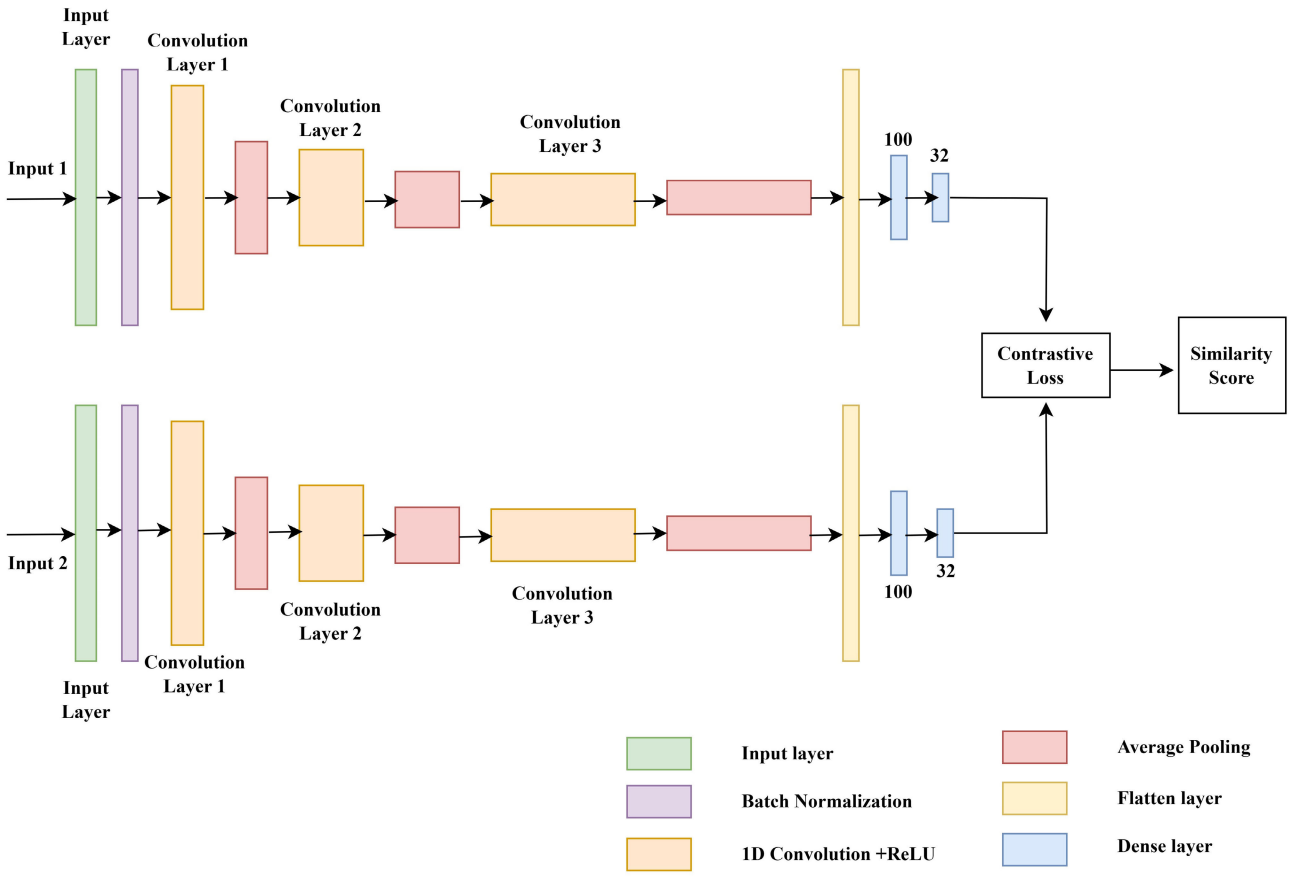

FIGURE 3. The neural network architecture for stage two.

TABLE 1. Topology and parameters of subnetworks of SiNN.

Layer	Configuration	Output Shape	Parameters
Input	none	256×3	0
Batch Normalization	none	256×1	4
Convolution 1D	Number of filters : 4 Kernal Size : 5 Activation : ReLU	252×4	24
Average Pooling 1D	Pool size : 2	126×4	0
Convolution 1D	Number of filters : 16 Kernal Size : 5 Activation : ReLU	122×16	336
Average Pooling 1D	Pool size : 2	61×16	0
Convolution 1D	Number of filters : 64 Kernal Size : 5 Activation : ReLU	57×64	5184
Average Pooling 1D	Pool size : 2	28×64	0
Flatten	none	1792	0
Dense	none	100	179300
Dense	none	32	3232

4) Employ the labeled pairs from Step 3 to train the proposed stage II neural network to create the differentiating model.

Retrieval Algorithm:

1) Create the latent space representations of the database images by utilizing the encoding model.

TABLE 2. Topology and parameters of SiNN.

Layer	Output Shape	Number of parameters
Input	256×1	0
Input	256×1	0
Functional	32	188080
Batch Normalization	1	4
Dense	1	2

TABLE 3. Showcases the average precision at rank K (K=1, 5, 10, 20, 50, and 100) for different classes in the dataset. It measures the average precision of retrieving the top-K images for each of the classes.

Image	AP@1	AP@5	AP@10	AP@20	AP@50	AP@100
Airplane	99.60	99.64	99.72	99.74	99.74	99.73
Automobile	99.79	99.75	99.72	99.71	99.69	99.68
Bird	99.70	99.68	99.72	99.71	99.74	99.74
Cat	100.00	99.86	99.86	99.85	99.82	99.83
Deer	100.00	99.87	99.83	99.85	99.83	99.82
Dog	99.46	99.42	99.41	99.46	99.43	99.42
Frog	99.70	99.68	99.70	99.67	99.65	99.68
Horse	99.70	99.74	99.79	99.76	99.74	99.71
Ship	99.70	99.63	99.56	99.61	99.61	99.57
Truck	99.89	99.73	99.76	99.77	99.76	99.76
Overall	99.75	99.7	99.71	99.71	99.7	99.69

TABLE 4. The table compares Mean Average Precision (mAP) values for existing techniques and our proposed approach. The mAP scores are evaluated for top-K image retrieval, where K represents the number of retrieved images. The values of K considered in this evaluation are 1, 5, 10, and 20. The mAP (Mean Average Precision) for image retrieval is calculated by averaging the Average Precision (AP) values for all query images at a given parameter K; we denote this value as mAP@K.

Approach[5]	mAP@1	mAP@5	mAP@10	mAP@20
ASPP-Net[1]	73.22	67.45	61.42	55.26
SSPP-Net[2]	86.64	80.53	75.36	68.18
TSPP-Net[3]	90.60	85.77	81.50	72.67
Improved TSPP-Net[4]	92.54	89.49	84.76	76.34
Proposed Approach	99.75	99.70	99.71	99.71

[1]ASPP-Net : Alexnet + Spatial pyramid pooling network
[2]SSPP-Net : Siamese + Spatial pyramid pooling network
[3]TSPP-Net : Triplet spatial pyramid pooling network
[4]Improved TSPP-Net : Improved Triplet + spatial pyramid pooling network
[5]The methodology and performance evaluation of the ASPP-Net, SSPP-Net, TSPP-Net, and Improved TSPP-Net approaches were presented in [13] by Yuan, Xinpan, et al.

2) Pass the query image to the encoding model to obtain the latent space representation of the query image.
3) Feed the latent space representation of the query image as one of the inputs into the differentiating model and simultaneously input the latent space representations of the database images as the other input. The model output will be a similarity score, indicating the level of similarity between the query image and the database images.
4) Rank the images based on the outputs from Step 3. Furnish the top-K images as the output of the CBIR system.

IV. EXPERIMENTAL RESULTS

The proposed methodology was evaluated using one of the public datasets. The details of the dataset used, the definitions of the performance metrics employed, and finally, the outcomes of the experiments done are described in this part.

A. DATASET

The dataset used for evaluation is the CIFAR-10 dataset [28]. This is one of the public datasets, with 60,000 images distributed over 10 different classes. The various classes present in the dataset are deer, truck, horse, bird, frog, automobile, dog, ship, cat, and airplane. The dimensions of the

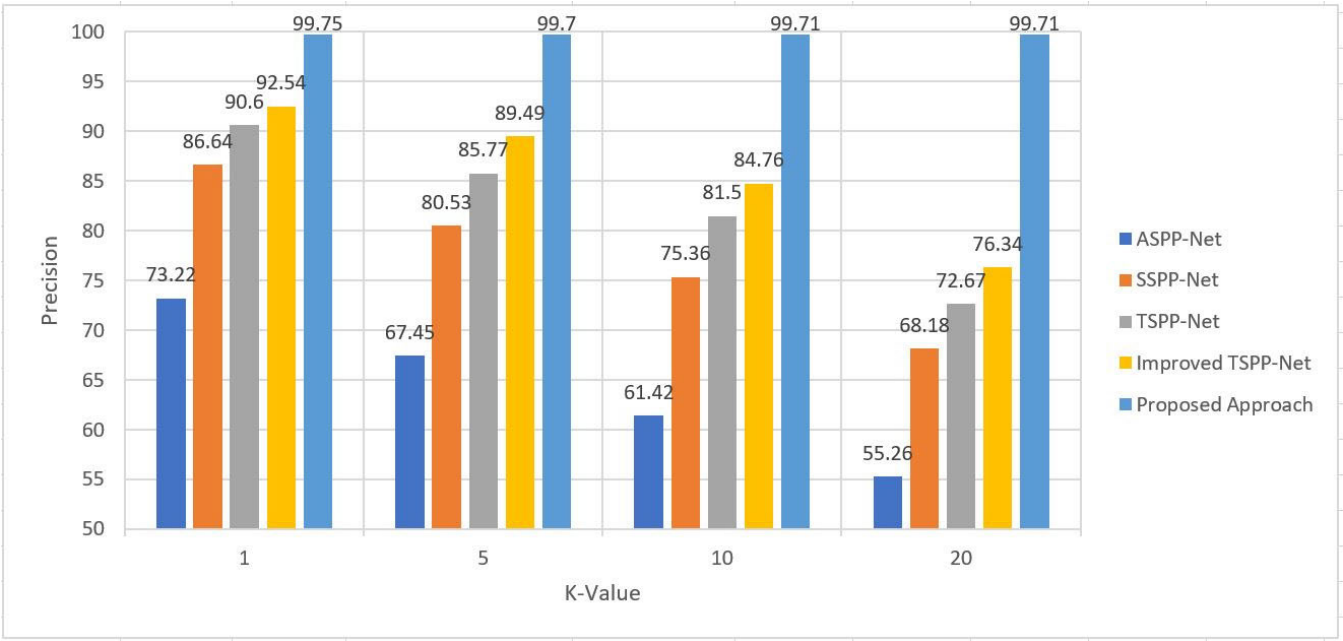

FIGURE 4. Performance comparison for different K-values for different state-of-the-art methods from [13] and the proposed method. Here K denotes the query input used to retrieve the top-K similar images.

TABLE 5. Mean Average Precision (mAP) is the measure used for comparison, and it is calculated by aggregating the values for Average Precision (AP) for each query image to retrieve the 20 most relevant images. The metric is denoted as mAP@20 to signify the number of images to be retrieved as a query response.

Method reference[1]	Approach used	mAP@20
Huang et al. [17]	Salient Coding	84.29
Camlica et al. [18]	Local Binary Patterns (LBPs)	86.80
Wu et al. [19]	Group Saliency Coding	91.07
Shamna et al. [20]	Bag of Visual Words	97.22
Öztürk [16]	CNN and AutoEncodeer	99.12
Proposed Method	SSiNN	99.71

[1]The performance measurements shown in the table for the existing techniques are taken from [16].

images in the dataset is 32 × 32. For the experiments, the number of images chosen was 10000, almost equally distributed for all the 10 different classes. The data distribution adheres to an 80:20 ratio, allocating 80% of the data for training the model and reserving 20% for testing and evaluating its performance. This partitioning scheme ensures a substantial amount of data is utilized for training purposes while reserving an independent subset specifically for comprehensive testing and rigorous assessment of the model's effectiveness.

B. PERFORMANCE METRICS

The success of a content-based image retrieval system can be evaluated based on how well it retrieves the intended images and rejects those that are irrelevant. One such metric used is Precision.

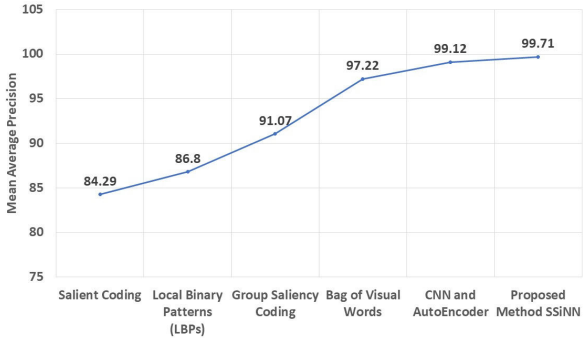

FIGURE 5. Performance comparison of P@20 for different state of art methods from [16] and proposed method.

Definition 2: The precision of a CBIR system is measured by how well it selects only the most pertinent images

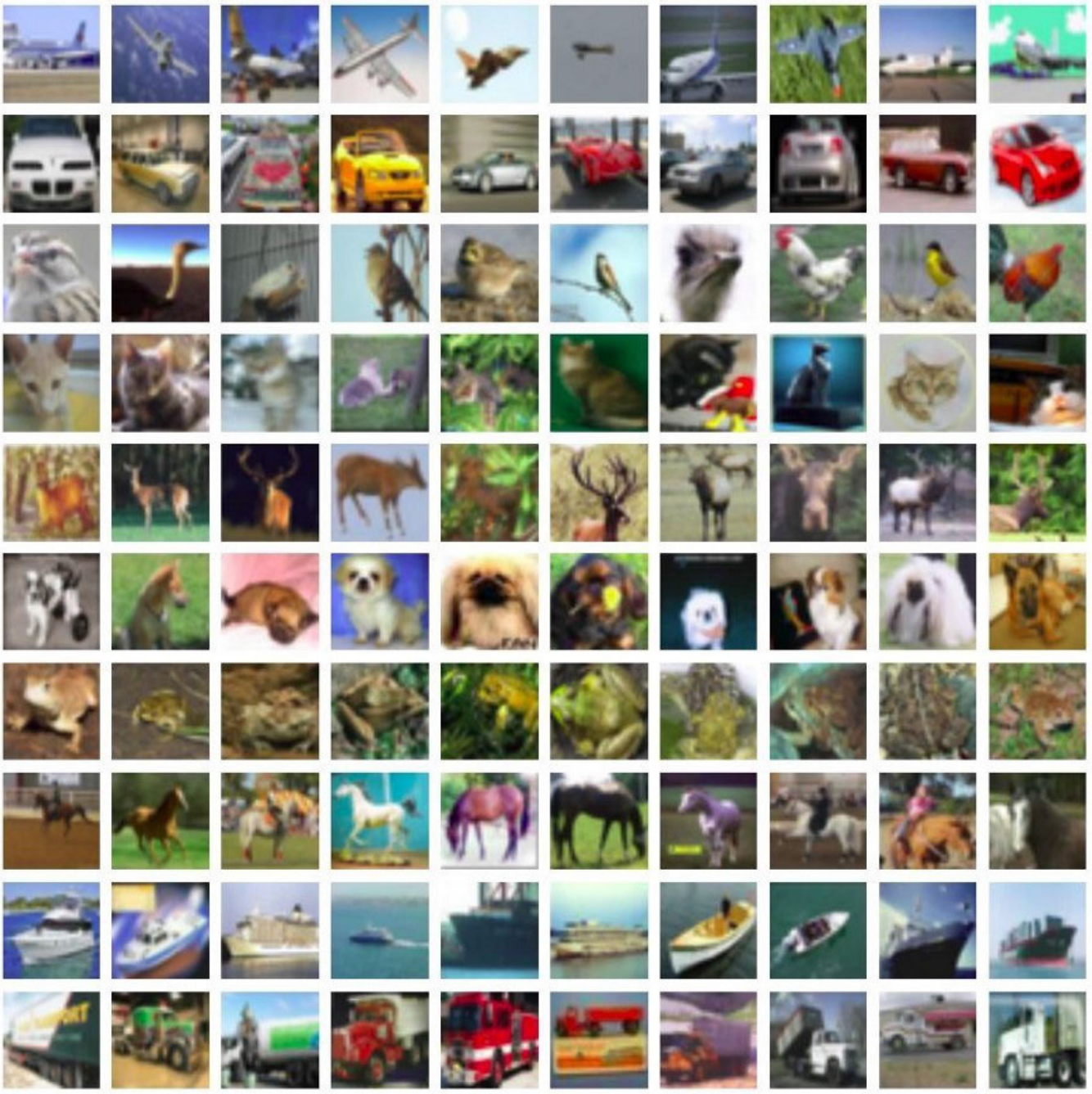

FIGURE 6. Sample images from CIFAR-10 dataset.

for retrieval.

$$Precision = \frac{Number\ of\ relevant\ images\ retrieved}{Total\ number\ of\ images\ retrieved} \quad (5)$$

Definition 3: Precision at rank K (P@K): It assesses the system's capacity to retrieve only relevant Images when the number of images retrieved is K.

$$P@K = \frac{Number\ of\ relevant\ images\ retrieved}{K} \quad (6)$$

Definition 4: Average Precision at rank K (P@K): Average Precision at a rank K is calculated by taking the average of P@K values across a particular class of images.

$$AP@K = \frac{\sum_{I \in C} P@K(I)}{|C|} \quad (7)$$

here C indicates the set of query images, which belong to a single category and |C| indicates the cardinality of the set C.

TABLE 6. The Mean Average Precision (mAP) of the SSiNN approach for the retrieval of top-K similar images is presented for different values of K.

K-value	mAP
1	99.75
5	99.70
10	99.71
20	99.71
50	99.70
100	99.69

Definition 5: The Mean average precision (mAP): Mean of AP@K, calculated across all the query images.

$$mAP@K = \frac{\sum_{n=1}^{N} AP@K(n)}{N} \tag{8}$$

here N indicates the number of distinct categories of query images present in the entire set of query images.

C. PERFORMANCE COMPARISION

The proposed CBIR framework's performance was assessed, and the performance metrics for extracting K-Images with varied values of K are shown in Table 3. We use mean average precision (mAP) as a metric to compare the performance of the proposed method to that of the existing ones. The CBIR system requires two input parameters: a query image and a numeric value K denoting the number of images to be retrieved. At a specific fixed K-value, we input query images from the test dataset into the proposed CBIR system. We then evaluate the precision values for the individual queries by comparing the retrieved images to the ground truth. We compute the mean average precision (mAP) for the given K-value by aggregating the individual precision values obtained from the evaluation process using Equations 7 and 8. This mAP@K metric provides a comprehensive assessment of the overall retrieval performance, considering the precision achieved across multiple queries.

Table 4 exhibits the performance of the proposed method in comparison with state-of-the-art methods by evaluating mAP for the retrieval of 1, 5, 10, and 20 relevant images. Table 5 and Figure 5 present the results of analyzing retrieval performance compared to another set of existing approaches for K = 20. In terms of performance, the comparison shows that the suggested method outperforms the existing methods. The mean average precision (mAP) for different K-values is presented in Table 6. We observe that the retrieval performance hasn't degraded with an increased K-value.

V. CONCLUSION

In this paper, we introduce a powerful content-based image retrieval algorithm that can be applied to any dataset. We take a two-staged approach to content-based image retrieval task, first constructing a latent space representation and then applying a deep learning-based image differentiating strategy based on a Siamese network architecture. Although the CIFAR-10 dataset was used for the described experiments, the framework is easily adaptable to other datasets. The outcomes prove that the method outperforms the current options.

The proposed model exhibits two key limitations. Firstly, it requires a substantial amount of labeled training data, specifically pairs of images with corresponding similarity labels, which can be a laborious and costly process, particularly when working with extensive image datasets. Secondly, the interpretability of the model is constrained, particularly in the second stage, where a Siamese neural network is employed to transform the initial feature representations. Understanding and interpreting the acquired representations and similarity metrics can present challenges, impeding effective debugging and enhancement of the model.

This paper presents a novel approach that tackles the content-based image retrieval (CBIR) problem by separating the encoding and differentiating stages. To showcase the concept's applicability, we employed straightforward architectures; however, there is room for experimentation with more advanced architectures, advanced loss functions, and hyper-parameter tuning to enhance the practical applications of CBIR in future work.

REFERENCES

[1] G. Khosla, N. Rajpal, and J. Singh, "Evaluation of Euclidean and Manhattan metrics in content-based image retrieval system," in *Proc. 2nd Int. Conf. Comput. Sustain. Global Develop. (INDIACom)*, 2015, pp. 12–18.
[2] T. Deselaers, D. Keysers, and H. Ney, "Features for image retrieval: An experimental comparison," *Inf. Retr.*, vol. 11, no. 2, pp. 77–107, Apr. 2008.
[3] S. R. Dubey, S. K. Singh, and R. K. Singh, "Rotation and illumination invariant interleaved intensity order-based local descriptor," *IEEE Trans. Image Process.*, vol. 23, no. 12, pp. 5323–5333, Dec. 2014.
[4] J. Wan, D. Wang, S. C. H. Hoi, P. Wu, J. Zhu, Y. Zhang, and J. Li, "Deep learning for content-based image retrieval: A comprehensive study," in *Proc. 22nd ACM Int. Conf. Multimedia*, 2014, pp. 157–166.
[5] H. Abdel-Nabi, G. Al-Naymat, and A. Awajan, "Content based image retrieval approach using deep learning," in *Proc. 2nd Int. Conf. New Trends Comput. Sci. (ICTCS)*, Oct. 2019, pp. 1–8.
[6] Z. Çamlica, H. R. Tizhoosh, and F. Khalvati, "Autoencoding the retrieval relevance of medical images," in *Proc. Int. Conf. Image Process. Theory, Tools Appl. (IPTA)*, Nov. 2015, pp. 550–555.
[7] D. Madhavi, K. M. C. Mohammed, N. Jyothi, and M. R. Patnaik, "A hybrid content-based image retrieval system using log-Gabor filter banks," *Int. J. Electr. Comput. Eng. (IJECE)*, vol. 9, no. 1, pp. 237–244, 2019.
[8] D. Madhavi and M. R. Patnaik, "Genetic algorithm-based optimized Gabor filters for content-based image retrieval," in *Proc. Intell. Commun., Control Devices (ICICCD)*. Singapore: Springer, 2017, pp. 157–164.
[9] D. Madhavi and M. R. Patnaik, "Image retrieval based on tuned color Gabor filter using genetic algorithm," *Int. J. Appl. Eng. Res.*, vol. 12, no. 15, pp. 5031–5039, 2017.
[10] A. Shakarami and H. Tarrah, "An efficient image descriptor for image classification and CBIR," *Optik*, vol. 214, Jul. 2020, Art. no. 164833.
[11] K. R. Kruthika, Rajeswari, and H. D. Maheshappa, "CBIR system using capsule networks and 3D CNN for Alzheimer's disease diagnosis," *Inform. Med. Unlocked*, vol. 14, pp. 59–68, 2019.
[12] S. Sabour, N. Frosst, and G. E. Hinton, "Dynamic routing between capsules," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 1–11.
[13] X. Yuan, Q. Liu, J. Long, L. Hu, and Y. Wang, "Deep image similarity measurement based on the improved triplet network with spatial pyramid pooling," *Information*, vol. 10, no. 4, p. 129, Apr. 2019.

[14] Y. Cai, Y. Li, C. Qiu, J. Ma, and X. Gao, "Medical image retrieval based on convolutional neural network and supervised hashing," *IEEE Access*, vol. 7, pp. 51877–51885, 2019.
[15] Y. Liu, L. Ding, C. Chen, and Y. Liu, "Similarity-based unsupervised deep transfer learning for remote sensing image retrieval," *IEEE Trans. Geosci. Remote Sens.*, vol. 58, no. 11, pp. 7872–7889, Nov. 2020.
[16] Š. Öztürk, "Stacked auto-encoder based tagging with deep features for content-based medical image retrieval," *Expert Syst. Appl.*, vol. 161, Dec. 2020, Art. no. 113693.
[17] Y. Huang, K. Huang, Y. Yu, and T. Tan, "Salient coding for image classification," in *Proc. CVPR*, Jun. 2011, pp. 1753–1760.
[18] Z. Çamlica, H. R. Tizhoosh, and F. Khalvati, "Medical image classification via SVM using LBP features from saliency-based folded data," in *Proc. IEEE 14th Int. Conf. Mach. Learn. Appl. (ICMLA)*, Dec. 2015, pp. 128–132.
[19] Z. Wu, Y. Huang, L. Wang, and T. Tan, "Group encoding of local features in image classification," in *Proc. 21st Int. Conf. Pattern Recognit. (ICPR)*, Nov. 2012, pp. 1505–1508.
[20] P. Shamna, V. K. Govindan, and K. A. Abdul Nazeer, "Content based medical image retrieval using topic and location model," *J. Biomed. Inform.*, vol. 91, Mar. 2019, Art. no. 103112.
[21] V. Rupapara, M. Narra, N. K. Gonda, K. Thipparthy, and S. Gandhi, "Auto-encoders for content-based image retrieval with its implementation using handwritten dataset," in *Proc. 5th Int. Conf. Commun. Electron. Syst. (ICCES)*, Jun. 2020, pp. 289–294.
[22] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, Nov. 2004.
[23] J. Sivic and A. Zisserman, "Video Google: A text retrieval approach to object matching in videos," in *Proc. 9th IEEE Int. Conf. Comput. Vis.*, 2003, pp. 1470–1477.
[24] J. Philbin, O. Chum, M. Isard, J. Sivic, and A. Zisserman, "Object retrieval with large vocabularies and fast spatial matching," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2007, pp. 1–8.
[25] F. Radenovic, G. Tolias, and O. Chum, "CNN image retrieval learns from BoW: Unsupervised fine-tuning with hard examples," in *Proc. 14th Eur. Conf. Comput. Vis. (ECCV)*, Amsterdam, The Netherlands: Springer, Oct. 2016, pp. 3–20.
[26] A. Gordo, J. Almazán, J. Revaud, and D. Larlus, "Deep image retrieval: Learning global representations for image search," in *Proc. 14th Eur. Conf. Comput. Vision (ECCV)*. Amsterdam, The Netherlands: Springer Oct. 2016, pp. 241–257.
[27] M. Li, X. Chen, X. Li, B. Ma, and P. M. B. Vitányi, "The similarity metric," *IEEE Trans. Inf. Theory*, vol. 50, no. 12, pp. 3250–3264, Dec. 2004.
[28] A. Krizhevsky and G. Hinton, "Learning multiple layers of features from tiny images," Univ. Toronto, Tech. Rep., 2009. [Online]. Available: http://www.cs.utoronto.ca/~kriz/learning-features-2009-TR.pdf

GOPU V. R. MUNI KUMAR received the B.Tech. degree in electronics and communications engineering from the Bapatla Engineering College, in 2005, and the M.E. degree in electrical communication engineering from the Indian Institute of Science, Bengaluru, India, in 2007. He is currently pursuing the Ph.D. degree with the GITAM School of Technology, Visakhapatnam, India. He is an Artificial Intelligence Solution Architect with Group 42. His research interests include data science, computer vision, security systems, and game theory.

D. MADHAVI received the A.M.I.E. degree, in 2000, and the M.Tech. and Ph.D. degrees from Andhra University, in 2004 and 2018, respectively. She has 21 years of teaching and research experience. She is an Associate Professor with the GITAM School of Technology, Visakhapatnam. Her research interests include image processing, VLSI, signal processing, and neural networks. She secured the Suman Sharma Award for the highest total in A.M.I.E.