Computers, Materials & Continua
DOI: 10.32604/cmc.2022.027720
Article

Tech Science Press

CNN Based Multi-Object Segmentation and Feature Fusion for Scene Recognition

Adnan Ahmed Rafique[1], Yazeed Yasin Ghadi[2], Suliman A. Alsuhibany[3], Samia Allaoua Chelloug[4,*], Ahmad Jalal[1] and Jeongmin Park[5]

[1]Department of Computer Science, Air University, Islamabad, 44000, Pakistan
[2]Department of Computer Science and Software Engineering, Al Ain University, Al Ain, 15551, UAE
[3]Department of Computer Science, College of Computer, Qassim University, Buraydah, 51452, Saudi Arabia
[4]Department of Information Technology, College of Computer and Information Sciences, Princess Nourah bint Abdulrahman University, P.O. Box 84428, Riyadh, 11671, Saudi Arabia
[5]Department of Computer Engineering, Korea Polytechnic University, Siheung-si, Gyeonggi-do, 237, Korea
*Corresponding Author: Samia Allaoua Chelloug. Email: sachelloug@pnu.edu.sa
Received: 25 January 2022; Accepted: 11 May 2022

Abstract: Latest advancements in vision technology offer an evident impact on multi-object recognition and scene understanding. Such scene-understanding task is a demanding part of several technologies, like augmented reality-based scene integration, robotic navigation, autonomous driving, and tourist guide. Incorporating visual information in contextually unified segments, convolution neural networks-based approaches will significantly mitigate the clutter, which is usual in classical frameworks during scene understanding. In this paper, we propose a convolutional neural network (CNN) based segmentation method for the recognition of multiple objects in an image. Initially, after acquisition and preprocessing, the image is segmented by using CNN. Then, CNN features are extracted from these segmented objects, and discrete cosine transform (DCT) and discrete wavelet transform (DWT) features are computed. After the extraction of CNN features and computation of classical machine learning features, fusion is performed using a fusion technique. Then, to select the minimal set of features, genetic algorithm-based feature selection is used. In order to recognize and understand the multi-objects in the scene, a neuro-fuzzy approach is applied. Once objects in the scene are recognized, the relationship between these objects is examined by employing the object-to-object relation approach. Finally, a decision tree is incorporated to assign the relevant labels to the scenes based on recognized objects in the image. The experimental results over complex scene datasets including SUN Red Green Blue-Depth (RGB-D) and Cityscapes' demonstrated a remarkable performance.

Keywords: Convolutional neural network; decision tree; feature fusion; neuro-fuzzy system

1 Introduction

Success in vision technologies for multi-object detection or scene recognition mainly depends upon the segmentation and key feature extraction techniques. Semantic segmentation nowadays has become an important technique for scene understanding by assigning a specific label to each pixel in a scene. There are various applications that use semantic segmentation with the combination of deep learning, such as handwritten stuff recognition, image categorization, finding objects in complex scenes, and driver assistance systems.

Researchers have been working on semantic segmentation, multi-object detection, and scene recognition for the last couple of decades. They are trying to deal with the challenges faced during object detection, background subtraction, features engineering, features optimization, computational time, and accurate scene classification. There are various feature extraction-based frameworks that are proposed by researchers to recognize objects and classify complex scenes. Some of the examples are color features, shape features, local binary patterns, scale-invariant feature transform (SIFT) features, binary robust invariant scalable keypoints (BRISK), and a bag of features. Additionally, a feature fusion technique is useful, and it can be applied in a variety of ways including parallel feature fusion, serial feature fusion, low- and mid-level feature fusion, and transfer-based fusion. These feature fusion-based methods have a significant role to improve the classification accuracies of existing systems.

Deep learning is an excellent candidate for this domain for its legitimate ability to overcome the limitations of conventional algorithms of feature extraction, such as handcrafted techniques. Deep learning includes CNN, which is a subtype of deep architecture. Nowadays CNN is considered to be an integral part of computer vision tasks. A wide range of applications, including semantic segmentation [1], scene analysis [2], inferring relationships amongst multiple objects, autonomous driving, and tourist guides are incorporating CNN as a robust component. Modern machine learning algorithms have successfully outdated earlier techniques that are based on low-level cues. Moreover, deep learning has achieved incredible success in image classification, semantic segmentation, speech recognition, multiple object detection, and scene understanding. However, the most active area of research is the image and video analysis via pixel-wise labeling based on CNN architecture [1,3,4]. In this paper, to demonstrate the effectiveness of this research, we used the following datasets: Cityscape's, SUN RGB-D, and NYUD2. These complex and diverse datasets include thousands of images having a large number of classes. These datasets present a multitude of challenges, including changes in illumination, varying hues, occlusion, and object class similarities. To overcome these challenges, we propose a new method for multi-object recognition based on semantic segmentation and CNN feature extraction, along with Discrete Cosine Transform (DCT) and Discrete Wavelet Transform (DWT) features. The proposed method consists of multiple steps. Firstly, all the input images are analyzed for semantic segmentation by employing a CNN. Secondly, the segmented objects are considered for feature extraction including deep CNN features through pre-trained CNN models such as SegNet and VGG, as well as classical machine learning features, i.e., DCT and DWT features. After computation of features, parallel feature fusion is applied to get a complete fused feature matrix which is then forwarded to the genetic algorithm to select the most appropriate features for further processing. The selected features are then taken as input by the Neuro-fuzzy system (NFS) for the recognition of multiple objects in complex indoor and outdoor scenes. The subsequent sections describe the details of each step.

The key contributions of this research article are as follows:

- After semantic segmentation, a fusion of CNN features and classical machine learning features including DWT and DCT is incorporated to enhance the performance of object recognition.

- An optimized Genetic Algorithm (GA) based feature selection method is employed to acquire the maximum results using reduced features set.
- We devised an object-to-object relation (OOR) based scene recognition after the objects are recognized in the complex scenes.

The remaining part of the paper is organized as follows: Literature review is summarized in Section 2. Section 3 offers a vision of the methodology, including the proposed framework for object recognition. Experimental analysis on datasets with an overview of datasets is given in Section 4. Finally, the conclusion and future work is presented in Section 5.

2 Related Work

Numerous researchers have devoted their energies towards the field of scene recognition by employing different semantic segmentation and object detection techniques. We have studied literature regarding segmentation, object detection, labeling, image classification, scene understanding, and recognition in dynamic environments considering Red Green Blue (RGB) and depth data. We discussed semantic segmentation with classical machine learning and deep learning in the following subsections.

2.1 Multi-objects Semantic Segmentation via Machine Learning

Semantic segmentation is a promising trend, inspired by demanding datasets [5]. Before the introduction of deep networks, the most effective algorithms rely entirely on handcrafted features that classified pixels separately. To estimate the class probabilities of the input image, a patch is usually fed through a classifier such as Classification Tree [6,7] or AdaBoost [8]. A number of studies have exploited appearance-based features or SfM over the CamVid dataset which is comprised of complex road scenes. To improve the accuracy, a conditional random field (CRF) is used after smoothing. More recent approaches feature based on appearance or SfM and appearance [9] have been explored for the CamVid road scene understanding test [10]. These per-pixel noisy predictions (often called unary terms) from the classifiers are then smoothed by using a pair-wise or higher-order CRF to improve the accuracy. The focus of the most recent research involves unaries of excellent quality by predicting the label based on all pixels instead of the only central pixel. By employing this technique, the result of Random Forest [11,12] is improved. In another approach, popular hand-drawn characteristics are combined with Spatio-temporal super pixelization to improve accuracy. Integrating object detection outputs using classifier projections in a CRF framework is the best strategy for object detection and semantic scene understanding. Semantic segmentation must be enhanced based on the results of all of these approaches.

Numerous researchers consider color spaces an important cue for color image segmentation. Jurio et al. [13] compared multiple color spaces using cluster-based segmentation to focus on similar techniques. They included four color spaces: HSV, CMY, RGB, and YUV to determine the best color representation model. Although the HSV produced decent results, they achieved the highest accuracy against the CMY color model. Sinop et al. [14] describe their graph-cut algorithm for image segmentation separating the foreground object from the background. The technique considers the whole image with its morphological details for efficient segmentation. Beunestado et al. [15] proposed an image segmentation method that combines the statistical confidence interval with the standard Otsu technique to achieve improved segmentation results. They enhanced the image by their proposed method using a statistical confidence interval and then applied the Otsu algorithm which provided good results compared to the standard Otsu algorithm.

2.2 Multi-objects Semantic Segmentation via Deep Learning

Modern research has produced remarkably accurate solutions even in sophisticated and cluttered scenes. However, deep learning-based segmentation techniques, on the other hand, have two fundamental limitations. The first one is the loss of feature resolution caused by subsequent pooling layers and the second one is the scaling factor in real-time visuals. Some of the researchers proposed solutions to deal with these limitations. Using an encoder-decoder structure, Fully Convolutional Networks [16] seek to fix these challenges. The object description and spatial information are obtained by the decoder. U-Net [17] establishes skip links to connect the features of both the encoder and the decoder. SegNet [1] preserves and utilizes pooling indices in the decoder segment. DeepLab-v2 [18] provides a challenging spatial pyramid pooling method that incorporates multi-scale data from a parallel structure. The pyramid pooling module in PSPNet [19] enhances efficiency by extracting the global context information and by aggregating the different region-based contexts. After the atrous spatial pyramid pooling module, Deeplab-v3 + [20] links the decoder portion. The encoder extracts rich semantic information using atrous convolution, and the decoder recovers the object boundary. In [21], Rashid et al. used a deep learning architecture based on multi-layer deep features selection and fusion for object recognition. Their approach yielded accurate recognition using three steps, including two deep learning architecture elements, i.e., for the fusion of features, Deep Convolution Network for image recognition, and, Inspection V3 for feature extraction. Additionally, they molded parallel maximum covariance, and for the selection of best features, Logistic Regression controlled the Entropy Variance algorithm. Zia et al. [22] suggested a solution for object recognition using a deep CNN. They designed a hybrid 2D/3D CNN that used a pre-trained network. Furthermore, they train their CNN over a small RGB-D dataset. They combined the features extracted from both RGB and depth models, into their hybrid model, to produce more accurate results.

2.3 Scene Classification

Scene classification and recognition aimed at labeling scenes based on recognized object categories and the contextual relationship between those objects. Many researchers using traditional systems explored scene understanding and labeling. These traditional systems compute the different features to recognize the objects and classify scenes. In [19] Zhao et al. proposed Pyramid Scene Parsing Network (PSPNet) that takes an image as input and extracts the features by using a pre-trained CNN. Then, by using pyramids pooling, they accumulate visual features via a multiple-level pyramid. Finally, they predict scene labels with the help of a convolution layer. In [23] Hussain et al. proposed a hybrid mechanism that combines features based on both the classical and deep learning techniques. After making the database balanced, two classical features pyramid histogram of gradients (PHOG) and central symmetric local binary patterns (CS-LBP) are fused with the deep learning features, extracted from the pre-trained CNN model. To choose the best features, they used joint entropy with the K-nearest neighbor. J. Feng et al. proposed a probabilistic topic model that uses latent Dirichlet allocation (LDA) to extract the features to make scene semantic recognition. They used deeper training LDA for features training. Xia et al. [24] discussed discriminative patch representations using neural networks and designed a hybrid architecture in which they developed a semantic manifold on top of multi-scale CNN. Then, they employed Markov Random Fields (MRF) to concatenate various features such as multi scales and spatial relations for detecting appropriate scenes.

3 Material and Methods

In this section, we proposed a novel multi-object detection and scene recognition model that recognizes and labels multiple objects in the RGB, as well as depth images. Initially, an image is taken as input for segmentation and analysis. Then, a CNN is applied for the segmentation of multiple objects present in the image. Deep CNN features, as well as DWT and DCT features, are extracted from the segmented objects. After the extraction of classical and deep CNN features, these features are then fused using parallel feature fusion. The resultant feature vector is then analyzed for feature selection based on a genetic algorithm (GA). The selected features are fed to neuro-fuzzy where multiple objects in the image are detected and recognized. These detected objects are then analyzed for object-to-object relations by computing the probability scores of each object. Finally, the decision tree is employed to predict the scene label based on these relationships between the objects. Fig. 1 represents the detailed flow of our proposed model.

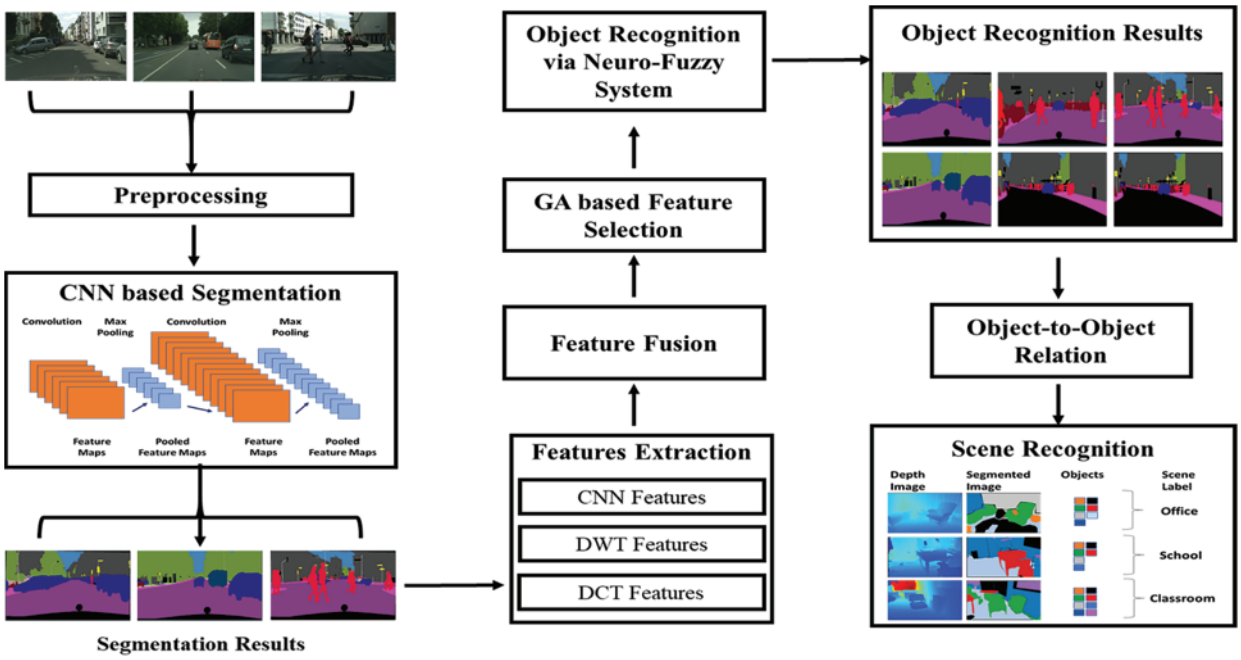

Figure 1: The system architecture of the proposed model representing the sequence of steps taken to recognize multiple objects and scene labels

3.1 Image Acquisition and Preprocessing

During pre-processing, un-sharp masking [25] for image sharpening is applied to get an enhanced image with clear edges. Un-sharp masking is performed by using three parameters: amount, radius, and threshold. The amount parameter is to control the contrast of the edges and is normally provided in percentage. Radius specifies the edge thickness and it may be enhanced. A Threshold is used to manage the brightness level of the image. During our experiments, we set the values of radius and amount parameters as 0.75% and 1.25% respectively. To get a sharpened image, Eq. (1) is applied.

$$I_{sh} = I_o + (I_o - I_b) * amt \tag{1}$$

where I_{sh} represents the sharpened image, I_o is to specify the original image, a blurred image is represented by I_b and *amt* is to describe the amount parameter. Fig. 2 demonstrates some of the examples of sharpening filter by using unsharp masking. To convolve the image, the following sharpening filter is used:

$$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix}$$

The above matrix is derived from Eq. (1) by using a uniform kernel having 5 pixels for I_b and 5 as a multiplier for *amt* parameter. The effect of the sharpening may be controlled by changing the value of the multiplier.

Figure 2: Few examples of Cityscape's dataset (a) Original images (b) Pre-processed images

3.2 Objects Segmentation

In this section, a comprehensive description of single/multi-object segmentation is introduced. Segmenting an image is to partition it into appropriate small regions. These small regions or segments are more meaningful and understandable to a machine for further processing. As with most complex images, there are usually several regions and objects in complex scenes, thus segmentation is a demanding yet critical process for accurate object recognition. Therefore, the quality of segmentation directly impacts the accuracy of object recognition. To improve the quality of segmentation, numerous researchers have adopted several different techniques for 2D and 3D object detection and recognition, like edge-based, region growing, model fitting, hybrid, and machine learning approaches.

A remarkable improvement in the results is witnessed with the advances in CNNs even in intricate and challenging scenarios [26]. In this paper, we applied a pre-trained CNN for the segmentation of multiple objects in complex scenes. The network is comprised of an encoder and decoder. The convolutional layers of the encoder are equivalent to the layers in the pre-trained network designed for semantic segmentation. Therefore, the initial weights for training purposes are used from a pre-trained network on large datasets [27]. The number of parameters also decreased as compared to other existing architectures. As every encoder layer corresponds to a decoder layer, hence, both the encoder and the decoder have the same 13 layers. Each encoder incorporates the convolution technique to generate a feature map. Before applying the rectified linear unit (ReLU) max (0, x), these feature maps are batch normalized. Then, Max-pooling having a 2×2 window and stride 2 is performed. The obtained result

is sub-sampled and feature maps are stored based on Max-pooling indices. These max-pooling indices are then utilized for the up-sampling process. As a result, sparse feature maps are generated. The sparse maps are convolved with filters that produce dense feature maps. The final output from the decoder is forwarded to the SoftMax classifier that computes the probabilities of pixels and assigns them unique class labels based on the maximum computed probability. CNN-based semantic segmentation findings are shown in Figs. 3 and 4.

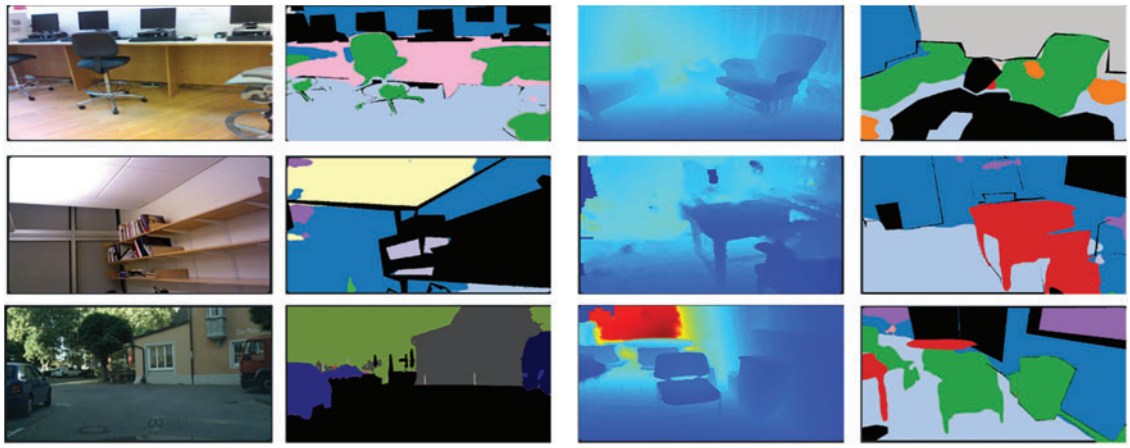

Figure 3: Segmentation of RGB-D images, (a) for RGB images using CNN over SUN-RGB-D dataset, (Left) RGB image, (Right) segmented multiple objects (b) for depth images using CNN over SUN-RGB-D dataset, (Left) depth image, (Right) segmented multiple objects

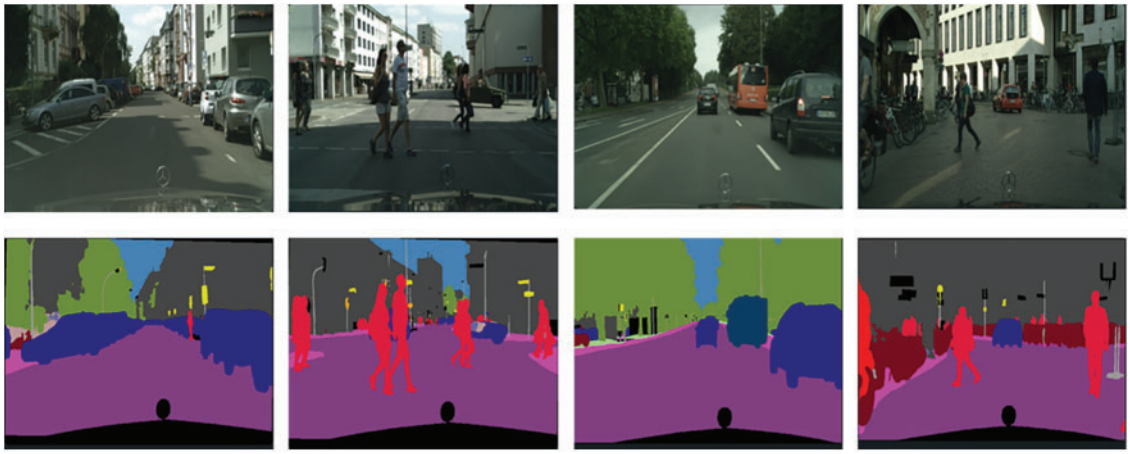

Figure 4: CNN-based Segmentation results over some images from city Cityscape's dataset (row1) original images (row2) segmented images

3.3 Feature Extraction Computation Over Segmented Objects

To recognize the objects in the scene, different features are computed, including deep and classical machine learning-based features. Deep learning-based include deep CNN features and classical

machine learning-based features are comprised of DCT and DWT features. The detailed feature computation, fusion, and selection process are described in the subsequent sections.

3.3.1 CNN Features Extraction Over Segmented Objects

To extract CNN features, Visual Geometry Group-16 (VGG-16) a pre-trained CNN model is incorporated. Deng et al. [28] trained this model on the ImageNet dataset. The model is simple and comprised of an input layer and thirteen convolutional layers. The input layer considers the images with a dimension of 320 × 320 × 3 as input. There are also five pooling layers that are using max-pooling concept followed by three fully connected layers. The window size for max-pooling is 2 × 2. The rectified linear unit (ReLU) is used as an activation function in hidden layers. To extract effective CNN features, a transfer learning method is applied that exploits the already learned features to make model useful as compared to a new model.

3.3.2 Discrete Cosine Transform (DCT) Over Segmented Objects

Many computer vision tasks are performed earlier by applying the DCT technique. We have performed the following steps of [29] to incorporate the DCT. Initially, we converted the RGB images to YCbCr. Secondly, block-wise (8 × 8 pixel) DCT is performed. Then, the DCT coefficients are quantized by employing a quantization matrix. All channels, i.e., Y, Cb, and Cr are subdivided into a group of 8 × 8 pixels. The value of each pixel is then preserved after subtracting 128 from the original value. Eq. (2) represents the DCT feature extraction as follows:

$$F_{DCT_{(u,v)}} = \frac{1}{4} a_u a_v \sum_{x=0}^{7} \sum_{y=0}^{7} g_{x,y} \cos\left[\frac{(2x+1)\, u\pi}{16}\right] \cos\left[\frac{(2y+1)\, v\pi}{16}\right] \tag{2}$$

where normalizing factors are represented by α_u and α_v, $g_{x,y}$ is the pixel value at (x, y), $F_{DCT\,(u,v)}$ is the DCT coefficient at (u, v), and $0 \leq u, v < 8$. The pixel information, in the DCT domain, is denoted by spatial frequency spectrums. Low-frequency sub-bands are observed on the upper-left of each 8 × 8 block, and high-frequency sub-bands are located on the bottom right. The frequency coefficients of lossy compression are quantized and converted to integers. The final quantization is performed using run-length coding and Huffman coding. Fig. 5 illustrates the DCT features of the different blocks from RGB-D Scenes dataset.

3.3.3 Discrete Cosine Transform (DCT) Over Segmented Objects

The DWT is a technique that empowers the process of image analysis while the image has multiple resolutions. Therefore, in order to classify multi-objects from complex scenarios, strong wavelet-based features are needed to be extracted as [30]. We can describe these features mathematically by using Eqs. (3) and (4) as follows:

$$W_{\varphi}\,(j_0, m, n) = \frac{1}{\sqrt{MN}} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f\,(x, y)\, \varphi_{j_0}, m, n\,(x, y) \tag{3}$$

$$W_{v}\,(j, m, n) = \frac{1}{\sqrt{MN}} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f\,(x, y)\, \psi_{j,m,n}\,(x, y) \tag{4}$$

where $\varphi\,(x, y)$ is a function to denote scaling while $\Psi\,(x, y)$ is a function to illustrate wavelets.

During the DWT feature computation, we decomposed our images from datasets up to level 2 which results in approximation and detailed coefficients. The approximation coefficients convey

valuable information regarding images. Therefore, we only considered the approximation coefficients. The Daubechies Wavelet 'db1' is incorporated as a Wavelet Function. Fig. 6 demonstrates the DWT features from Cityscape's dataset. The approximation coefficients are computed for an image of 64 × 64 up to two levels of decomposition, L1, and L2 respectively. These approximation coefficients are used as feature vectors having dimensions of 1024 × 1 at L1 and 256 × 1 at L2. Therefore, we can represent the feature vector by formulating Eq. (5) as follows:

$$\text{Feature}_{DWT} = \left[Apx_{\text{coefficient}_{(L1)}} \quad Apx_{\text{coefficient}_{(L2)}} \right] \tag{5}$$

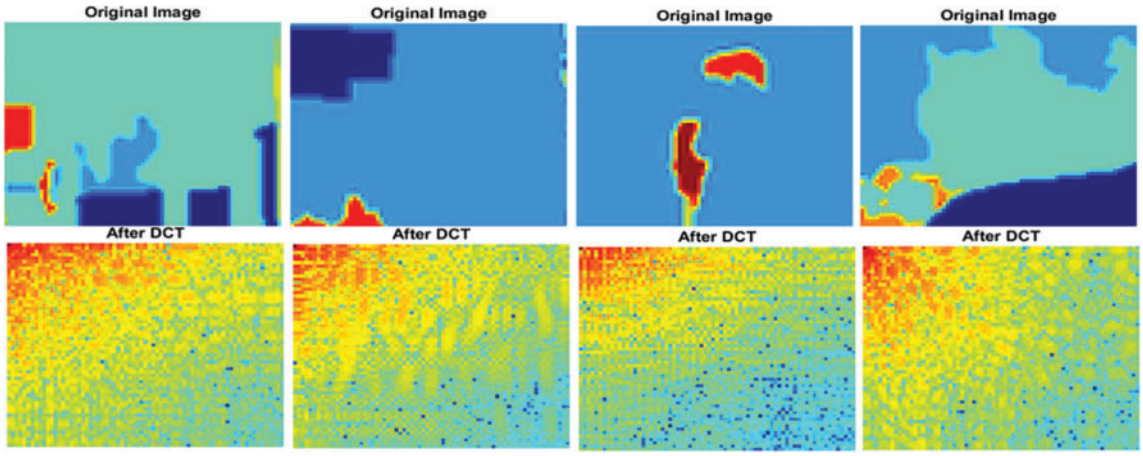

Figure 5: Feature extraction using DCT. (Upper) Patches of original images from SUN RGB-D dataset (Lower) features computation using DCT from the fixed size patches

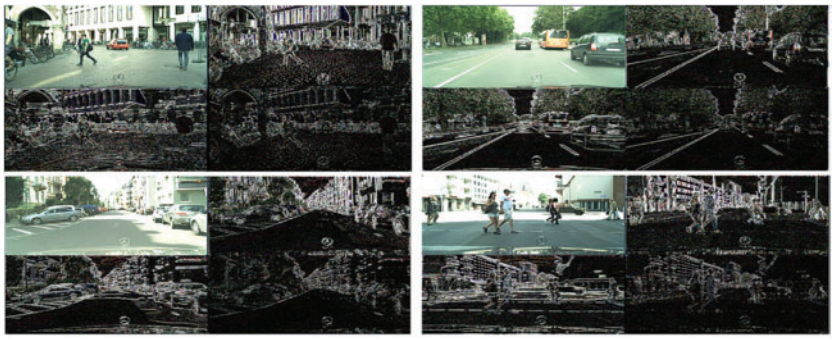

Figure 6: Feature's extraction using DWT technique

3.4 Feature Fusion and GA Based Feature Selection

The CNN, DCT, and DWT features are computed separately as Feature_{CNN}, Feature_{DCT}, and Feature_{DWT} respectively. All these feature vectors are merged to form a complete fused feature vector and normalized before fusion, to ensure that the individual feature vector elements do not surpass other elements. Once normalization is performed, the CNN, DCT, and DWT features are combined to form a complete fused feature vector and represented by Eq. (6).

$$\text{Feature}_{\text{Fused}} = \left[\text{Feature}_{CNN} \quad \text{Feature}_{DCT} \quad \text{Feature}_{DWT} \right] \tag{6}$$

A high-dimensional feature vector is obtained as a result of the two-level decomposition of complex images while DWT analysis is executed. Consequently, an inadequate classification is witnessed when the input feature vectors have high dimensions. Therefore, reducing the size of feature vectors is important in order to reduce computational costs and improve performance. For this purpose, a GA-based [31–33] features selection is employed to get a reduced dimensional feature vector $\text{Feature}_{\text{Final}}$ in Eq. (7).

$$\text{Feature}_{\text{Final}} = GA\ \{\text{Feature}_{\text{Fused}}\} \tag{7}$$

As a result of a combination of CNN, DCT, and DWT analysis, the performance of the classification is improved. The best performance of the classifier is used to determine the dimensions of the fused feature vector. We considered the dimensionality of 80. For a 64×64 size image, we used GA-based dimensionality of 50 for feature fusion this maximizes the classification accuracy.

3.5 Multi-Object Recognition via Neuro-Fuzzy System (NFS)

The fused feature vector after GA-based feature selection is fed to a Neuro-fuzzy classifier [34] for the recognition of multiple objects based on features extracted in the previous subsections. The architecture of the neuro-fuzzy system as presented in Fig. 7, reveals that it consists of input, output, and a hidden layer having perceptrons. The input layer takes the feature vectors while the output layer contains multiple perceptrons depending on the number of object classes in the training dataset.

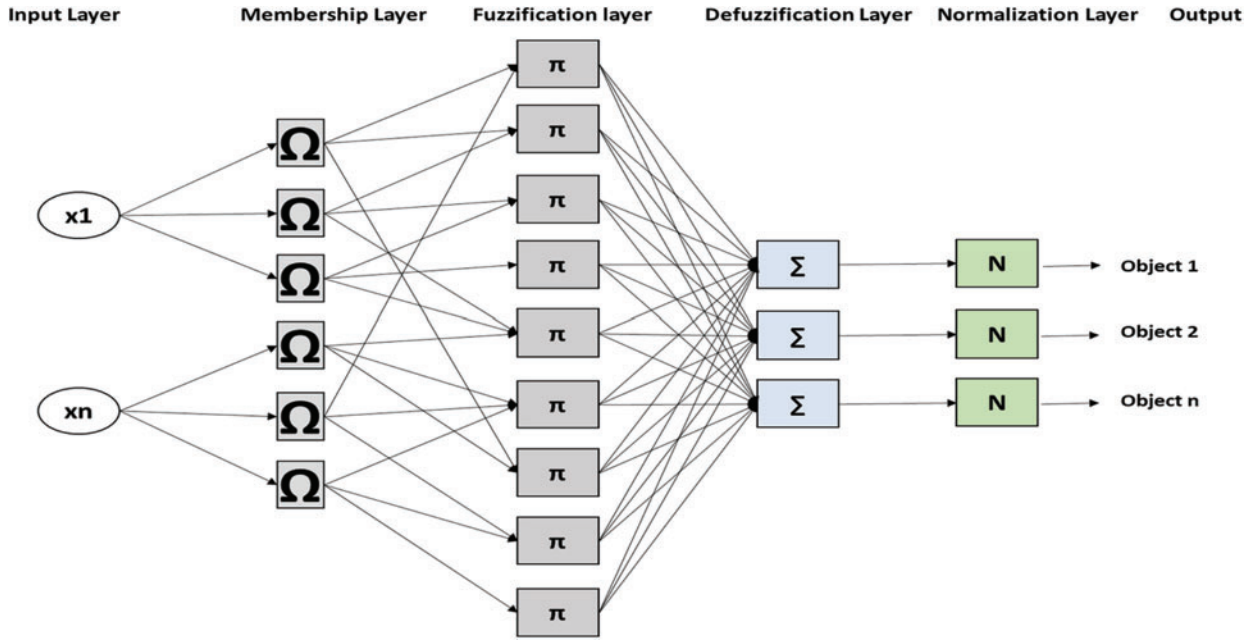

Figure 7: Flow of neuro-fuzzy object detection system for proposed scene recognition model

Since the consequence parameter is a linear equation, the initial component of NFS is based on the Takagi-Sugeno-Kang (TSK) approach of the first order. As well as n input nodes $X_1 - X_n$, each had numerous TSK rules and a single output variable. Eq. (8) denotes each of the NFS fuzzy rules takes the form of IF-THEN:

$$\text{Rule } N: \text{ IF } x_i \text{ is } A_i^N \text{ and } x_{i+1} \text{ is } A_{i+1}^N, \text{ THEN } y_N = a_0^N + \sum_{j=1}^{n} a_j^N x_j \tag{8}$$

where the membership function is represented by A_i^N, inputs are denoted by x_i, and the parameters of the consequent equation are described as a_j^N . The reasoning system for the TSK model consists of five levels, each of which has the following functions:

3.5.1 Input Layer

The nodes follow a form where variables and functions take the form of rule bases as described by TSK. Every node performs a calculation of a membership value. For determining the degree of membership value, which is defined in terms of the Gaussian function and is employed as Eq. (9).

$$\mu_{A_i}(x_i) = \exp\left[-\frac{(x_i - c_i)^2}{2\sigma_i^2}\right] \tag{9}$$

where c_i and σ_i are respectively the mean (or center) and the variance (or width) of the Gaussian membership function of the x_i input variable.

3.5.2 Membership Layer

This layer is carrying out the algebraic product operation for all the functions produced by the previous layers. w_i is the output of this layer and is an indication of the strength of the process. It is denoted by Eq. (10).

$$w_i = \prod_{j=1}^{n} \mu_{A_j}(x_j) \tag{10}$$

3.5.3 Fuzzification Layer

This layer constitutes fixed nodes that calculate the ratio of the detection strength, to the sum of all expression's strengths. The normalized detection strength is given by Eq. (11).

$$\overline{w_i} = \frac{w_i}{\sum_{k=1}^{r} w_k} \tag{11}$$

where w_i is the total number of rules.

3.5.4 Defuzzification Layer

Every node in this layer is an adaptive node that calculates the consequence value given by Eq. (12).

$$\overline{w_i} y_N = \overline{w_i}\,(a_0 + a_1x_1 + a_2x_2 + \cdots + a_nx_n) \tag{12}$$

where w_i is the normalized detection strength from layer 3 and $(a_0 + a_1x_1 + a_2x_2 + \cdots + a_nx_n)$ is the parameters of these nodes.

3.5.5 Normalization Layer

It normalizes the data acquired from the defuzzification layer to prepare it for the input to the final output layer.

3.5.6 Output Layer

It includes a fixed node denoted as $\sum_i$ that functions as a summation of the overall NFS output network. By the Weighted Fuzzy Mean (WFM) method, the overall output is obtained by Eq. (13).

$$y = \sum_i \bar{w}_i y_N = \frac{\sum_i w_i y_i}{\sum_i w_i} \tag{13}$$

Fig. 8 shows the results of multiple object recognition by applying NFS over Cityscape's dataset.

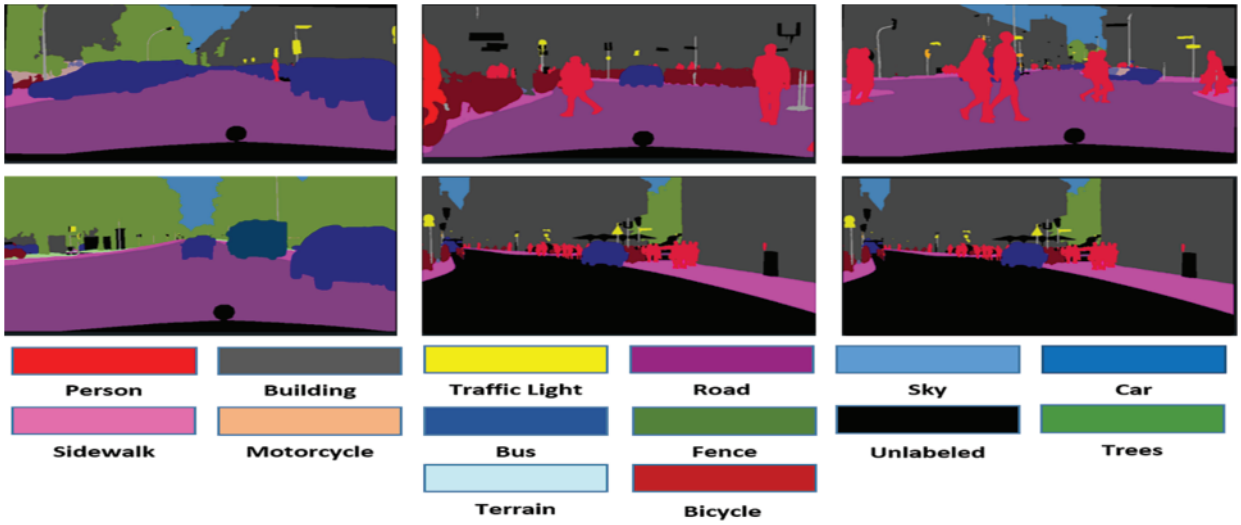

Figure 8: Recognition results over a few examples of Cityscape's dataset using a neuro-fuzzy technique

3.6 Object-to-Object Relations (OOR)

After multiple object recognition in a complex scene, the relationship between these objects is identified. To enhance the scene recognition performance, object-to-object relations (OOR) [32] are computed based on contextual information regarding objects. As complex scenes are comprised of multiple co-occurring visual features and the OOR significantly recognizes patterns to understand the scenes. For instance, a car is likely to be seen on roads instead of the sky or water. Similarly, an office table is likely to be in the office instead of on the roads. To determine the OOR, several features and the relative position of the object is considered. Initially, to find the weight of the target object j^{th} for $j \in \{1, 2, \ldots, n\}$ with the other relevant object i^{th} for $i \in \{1, 2, \ldots, n\}$, a dot product is computed by using Eq. (14) as follows:

$$w_t(j, i) = \frac{f_j \cdot f_i}{d(j, i)} \tag{14}$$

where the visual cures of j^{th} and i^{th} objects are represented by f_j and f_i respectively.

3.7 Scene Recognition

Once the OOR is determined, a decision tree is applied to recognize the scene label by incorporating the object class and contextual relationship between those objects. The decision tree model is comprised of multiple nodes, where each node tests a condition on every input and based on the

condition, each branch node represents the outcome of the test. The scene label is predicted at leaf nodes after computing all the necessary attributes. The classification rules are defined by the entire route from the root node to the leaf node. There are three types of nodes involved in the decision tree: Decisions, chance, and end nodes. Squares, circles, and triangles are normally used to describe a decision, chance, and end nodes respectively. Though complex ensemble models also produce good results, our results are highly accurate due to hyper-features along with decision trees. The use of hyper-features is simple because they simply correspond to the proposed version of the CLIQUE algorithm's basic characteristics. However, the other ensembles are more costly as each tree needs to fix the original classification. Fig. 9 demonstrate the scene recognition results over the Cityscape's dataset respectively.

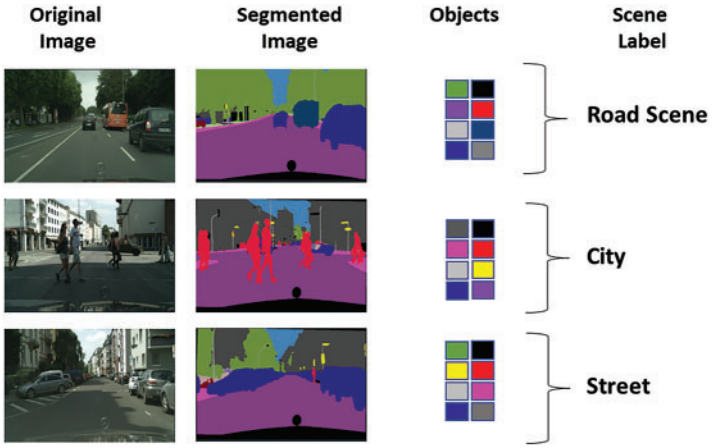

Figure 9: The results of proposed model for scene recognition over Cityscape's dataset

4 Performance Evaluation

Testing and validation of the proposed model are performed on three benchmark datasets: Cityscape, SUN RGB-D, and NYU-Dv2 datasets.

4.1 Datasets Description

The Cityscape's dataset [35] consists of 30 classes, including 50 cities that focus on urban street scenes for semantic scene understanding. The dataset is comprised of 5000 images of complex scenarios having different weather conditions, e.g., summer, spring, or fall. These images belong to one of the following classes: road, sidewalk, rail track, person, car, truck, bus, motorbike, bicycle, building, wall, fence, traffic sign, pole, traffic light, sky, or ground.

The SUN RGB-D dataset [36] is comprised of 10,355 RGB-D images and each image belongs to one of the 19 categories of SUN RGB-D. The dataset is a collection of RGB-D images from multiple datasets including NYU-Dv2, Berkeley B3DO, and SUN3D. Following settings are assumed to compare with state-of-the-art techniques: The training set consists of 5510 images while the testing set has 4845 images.

The NYUDv2 dataset [37] comprises 2347 labeled and 108, 617 unique unlabeled frames of 7 types with 64 different indoor scenes. These frames/scenes belong to the following seven types of classes: bathroom, bedroom, bookstore, cafe, kitchen, living room, and office. These classes consist of different

objects, including bed, bookshelf, book, cabinet, ceiling, floor, picture, sofa, table, TV, wall, window, background, unlabeled, etc.

4.2 Experimental Settings and Results

The experiments were performed on three publicly available datasets to determine the efficiency of the proposed model.

4.2.1 Recognition Accuracy

In this experiment, the three different features, i.e., CNN, DCT and DWT, were extracted and fused to apply for object recognition using the SUN RGB-D, Cityscape's, and NYUDv2 datasets. Tab. 1 demonstrates a confusion matrix over the Cityscape's dataset for object recognition. The average recognition accuracy of 96.16% is reported over the Cityscapes dataset that were executed with 25 iterations for the experiment. Average recognition accuracies of 63.1% and 72.8% were achieved over the SUN RGB-D and the NYUDv2 datasets while using a fusion of three features and 25 iterations as depicted in Tabs. 2 and 3 respectively.

Table 1: Recognition accuracy over Cityscape's dataset

Objects	bus	bcl	car	psn	rod	bdg	swk	mbk	trn	trs	sky	tre
Bus	**0.97**	0	0	0	0	0	0.03	0	0	0	0	0
Bcl	0	**0.96**	0	0	0.04	0	0	0	0	0	0	0
Car	0.02	0	**0.98**	0	0	0	0	0	0	0	0	0
Psn	0.01	0	0	**0.99**	0	0	0	0	0	0	0	0
Rod	0	0.06	0	0	**0.90**	0	0	0	0	0.02	0	0.02
Bdg	0	0	0	0	0	**0.93**	0	0	0.07	0	0	0
Swk	0.03	0	0	0	0	0	**0.93**	0.04	0	0	0	0
Mbk	0	0	0	0	0	0.03	0	**0.97**	0	0	0	0
Trn	0	0	0	0.02	0	0.07	0	0	**0.93**	0	0	0
Trs	0	0.05	0	0	0	0	0	0	0	**0.95**	0	0
sky	0	0.04	0	0	0	0	0	0	0	0	**0.96**	0
tre	0	0.06	0	0	0	0	0	0	0	0	0	**0.94**

Mean Accuracy = 96.16%

Note: Bus = bus; bcl = bicycle; car = car; psn = person; rod = road; bdg = building; swk = sidewalk; mbk = motor bike; trn = terrain; sky = sky; tre = tree

4.2.2 Comparison with State-of-the-art (SOTA) Methods

In this section, a comparison with other existing techniques is conducted. We observed an increase of a minimum of 2.25% accuracy over Cityscape's dataset, while a minimum of 6.4% and 3.6% increase in the accuracy of SUN-RGB-D and NYU-Dv2 datasets respectively is achieved during our experiments. Tab. 4 exploits the comparison of recognition accuracies over benchmark datasets with other SOTA techniques.

Table 2: Recognition accuracy over SUN-RGBD dataset

Objects Classes	chr	ctl	sof	tab	bow	cap	cbx	Cmg	scn	wal	flr
chr	**0.69**	0	0.31	0	0	0	0	0	0	0	0
ctl	0	**0.63**	0	0.26	0	0	0.11	0	0	0	0
sof	0.35	0	**0.65**	0	0	0	0	0	0	0	0
tab	0	0.18	0.20	**0.62**	0	0	0	0	0	0	0
bow	0	0	0	0.34	**0.57**	0	0	0	0	0.09	0
cap	0	0	0	0	0	**0.59**	0	0.22	0	0.19	0
cbx	0	0	0	0	0.13	0	**0.51**	0.19	0.17	0	0
cmg	0	0	0	0	0.12	0.21	0	**0.53**	0	0.14	0
scn	0	0	0	0	0	0	0.09	0.26	**0.67**	0	0
wal	0	0	0	0	0	0	0.13	0	0	**0.73**	0.14
flr	0	0.02	0	0.11	0	0	0	0	0	0.12	**0.75**

Mean Accuracy = 63.1%

Note: Chr = chair; ctl = coffee table; sof = sofa; tab = table; bow = bowl; cap = cap; cbx = cereal box; cmg = coffee mug; scn = soda can; wal = wall; flr = floor

Table 3: Recognition accuracy over NYU-Dv2 dataset

Objects	bed	bok	cab	cel	Flr	sof	tab	tvn	wal	win
bed	**0.75**	0	0	0	0.14	0	0.11	0	0	0
bok	0	**0.79**	0.07	0	0	0	0	0.14	0	0
cab	0.05	0	**0.69**	0	0.12	0	0	0	0.14	0
cel	0	0	0	**0.77**	0	0	0	0	0.15	0.08
flr	0.05	0	0	0	**0.76**	0	0.13	0	0	0.11
sof	0	0	0.12	0	0	**0.69**	0.12	0	0.07	0
tab	0.24	0	0.03	0	0	0	**0.73**	0	0	0
tvn	0	0.25	0	0	0	0	0	**0.75**	0	0
wal	0	0	0	0.17	0	0	0	0	**0.67**	0.16
win	0	0.13	0	0	0	0	0	0	0.19	**0.68**

Mean Accuracy = 72.8%

Note: bed=bed; bok=book; cab=cabinet; cel=ceiling; flr=floor; sof=sofa; tab=table; tvn=television; wal=wall; win=window

Table 4: Comparison of recognition accuracy (%) over benchmark datasets

Method	Cityscape's	SUN RGB-D	NYUv2
Khodabandeh et al. [38]	90.13	–	–
Wang et al. [39]	93.88	–	–
Song et al. [40]	–	–	66.9
Song et al. [41]	–	53.8	67.5
Xiong et al. [42]	–	56.2	68.1
Du et al. [43]	–	56.7	69.2
Proposed	**96.13**	**63.1**	**72.8**

5 Discussion

The proposed multi-object detection and scene recognition model is designed to achieve a state-of-the-art performance over RGB and RGB-Depth images. In this paper, semantic segmentation has a key role in the overall scene recognition process, however, our approach of the fusion of CNN, DCT, and DWT has significantly improved the overall accuracy of the scene recognition model. While, computational complexity is examined for Cityscape's as 1856.3 s, SUN RGB-D as 2543.9 s and NYUv2 as 3294.7 s. Initially, we only consider CNN features and performed the experiments for scene recognition. The results demonstrated an accuracy of 88.72%, 64.11%, and 58.25% over cityscapes', NYU-Dv2 datasets, and SUN RGB-D respectively. Then, we fused CNN features with DCT and conducted experiments for scene recognition. Improved accuracies of 91.55%, 68.91%, and 61.17% over cityscapes, NYU-Dv2, and SUN RGB-D respectively are observed.

6 Conclusion

In this paper, we introduced a novel scene recognition model to predict scene labels based on multiple objects recognition and OOR in different indoor and outdoor complex environments. Key achievements: like the segmentation of indoor scene depth objects, a combination of robust extraction of CNN and classical features for distinguishing each object, are attained in this study. The impact of the proposed model over the previous techniques is highlighted, with recognition accuracies of 96.16%, 63.1%, and 72.8% over Cityscape's, SUN RGB-D, and NYUDv2 datasets, respectively. Moreover, results suggest that our proposed technique is ideal for multi-object recognition against any change in the environment with consistent results and can be adopted in numerous applications, like medical diagnostics, video surveillance, robotic navigation, and indoor/outdoor scene understanding.

We are committed to extending our work towards point cloud-based object detection and scene recognition over depth images by incorporating deep learning techniques.

Funding Statement: This research was supported by a grant (2021R1F1A1063634) of the Basic Science Research Program through the National Research Foundation (NRF) funded by the Ministry of Education, Republic of Korea.

In addition; the authors would like to thank the support of the Deanship of Scientific Research at Princess Nourah bint Abdulrahman University. This work has also been supported by Princess

Nourah bint Abdulrahman University Researchers Supporting Project number (PNURSP2022R239), Princess Nourah bint Abdulrahman. University, Riyadh, Saudi Arabia.

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

References

[1] V. Badrinarayanan, A. Kendall and R. Cipolla, "Segnet: A deep convolutional encoder-decoder architecture for image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 12, pp. 2481–2495, 2017.

[2] M. Mandal and S. K. Vipparthi, "Scene independency matters: An empirical study of scene dependent and scene independent evaluation for CNN-based change detection," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, pp. 1–14, 2020.

[3] H. Noh, S. Hong and B. Han, "Learning deconvolution network for semantic segmentation," in *Proc. of the IEEE International Conference on Computer Vision*, Santiago, Chile, pp. 1520–1528, 2015.

[4] C. Farabet, C. Couprie, L. Najman and Y. LeCun, "Learning hierarchical features for scene labeling," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 8, pp. 1915–1929, 2012.

[5] S. Hao, Y. Zhou and Y. Guo, "A brief survey on semantic segmentation with deep learning," *Neurocomputing*, vol. 406, pp. 302–321, 2020.

[6] A. Jalal, I. Akhtar and K. Kim, "Human posture estimation and sustainable events classification via pseudo-2D stick model and K-ary tree hashing," *Sustainability*, vol. 12, no. 23, pp. 9814, 2020.

[7] A. Jalal, M. Batool and K. Kim, "Sustainable wearable system: Human behavior modeling for life-logging activities using K-ary tree hashing classifier," *Sustainability*, vol. 12, no. 24, pp. 10324, 2020.

[8] Y. Yue and Y. Yang, "Improved Ada boost classifier for sports scene detection in videos: From data extraction to image understanding," in *Proc. of Int. Conf. on Inventive Computation Technologies*, Coimbatore, India, pp. 1–4, 2020.

[9] A. Ahmed, A. Jalal and K. Kim, "Multi-objects detection and segmentation for scene understanding based on texton forest and kernel sliding perceptron," *Journal of Electrical Engineering and Technology*, vol. 16, no. 2 pp. 1143–1150, 2021.

[10] G. J. Brostow, J. Fauqueur and R. Cipolla, "Semantic object classes in video: A high-definition ground truth database," *Pattern Recognition Letters*, vol. 30, no. 2, pp. 88–97, 2009.

[11] C. Zhang, L. Wang and R. Yang, "Semantic segmentation of urban scenes using dense depth maps," in *Proc. of European Conf. on Computer Vision*, Berlin, Heidelberg, pp. 708–721, 2010.

[12] M. Javeed, A. Jalal and K. Kim, "Wearable sensors based exertion recognition using statistical features and random forest for physical healthcare monitoring," in *Proc. of Int. Bhurban Conf. on Applied Sciences and Technologies*, Islamabad, Pakistan, pp. 512–517, 2021.

[13] A. Jurio, M. Pagola, M. Galar, C. Lopez-Molina and D. Paternain, "A comparison study of different color spaces in clustering based image segmentation," in *Proc. of Int. Conf. on Information Processing and Management of Uncertainty in Knowledge Based Systems*, Sorrento, Italy, pp. 532–541, 2010.

[14] A. K. Sinop and L. Grady, "A seeded image segmentation framework unifying graph cuts and random walker which yields a new algorithm," in *Proc. of IEEE 11th Int. Conf. on Computer Vision*, Rio de Janeiro, Brazil, pp. 1–8, 2007.

[15] P. Buenestado and L. Acho, "Image segmentation based on statistical confidence intervals," *Entropy*, vol. 20, no. 1, pp. 46, 2018.

[16] W. Sun and R. Wang R, "Fully convolutional networks for semantic segmentation of very high resolution remotely sensed images combined with DSM," *IEEE Geoscience and Remote Sensing Letters*, vol. 15, no. 3, pp. 474–478, 2018.

[17] O. Ronneberger, P. Fischer and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. of Int. Conf. on Medical Image Computing and Computer-Assisted Intervention*, Munich, Germany, pp. 234–241, 2015.
[18] L. C. Chen, G. Papandreou, I. Kokkinos, K. Murphy and A. L. Yuille, "Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 4, pp. 834–848, 2017.
[19] H. Zhao, J. Shi, X. Qi, X. Wang and J. Jia, "Pyramid scene parsing network," in *Proc. of the IEEE Conference on Computer Vision and Pattern Recognition*, Honolulu, HI, USA, pp. 2881–2890, 2017.
[20] L. C. Chen, Y. Zhu, G. Papandreou, F. Schroff and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *Proc. of the European Conf. on Computer Vision*, Munich, Germany, pp. 801–818, 2018.
[21] M. Rashid, M. A. Khan, M. Alhaisoni, S. H. Wang, S. R. Naqvi *et al.,* "A sustainable deep learning framework for object recognition using multi-layers deep features fusion and selection," *Sustainability*, vol. 12, no. 12, pp. 5037, 2020.
[22] S. Zia, B. Yuksel, D. Yuret and Y. Yemez, "RGB-D object recognition using deep convolutional neural networks," in *Proc. of the IEEE Int. Conf. on Computer Vision Workshops*, Venice, Italy, pp. 896–903, 2017.
[23] N. Hussain, M. A. Khan, M. Sharif, S. A. Khan, T. Saba *et al.,* "A deep neural network and classical features based scheme for objects recognition: An application for machine inspection," *Multimedia Tools and Applications*, vol. 79, pp. 1–23, 2020.
[24] S. Xia, J. Zeng, L. Leng and X. Fu, "Ws-am: Weakly supervised attention map for scene recognition," *Electronics*, vol. 8, no. 10, pp. 1072, 2019.
[25] S. Lin, C. Wong, G. Jiang, M. Rahman, T. Ren *et al.,* "Intensity and edge based adaptive unsharp masking filter for color image enhancement," *International Journal for Light and Electron Optics*, vol. 127, no. 1, pp. 407–414, 2016.
[26] T. Akilan, Q. J. Wu, A. Safaei, J. Huo and Y. Yang, "A 3D CNN-LSTM-based image-to-image foreground segmentation," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 3, pp. 959–971, 2019.
[27] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh *et al.,* "Imagenet large scale visual recognition challenge," *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211–252, 2015.
[28] J. Deng, W. Dong, R. Socher, L. J. Li, K. Li *et al.,* "Imagenet: A large-scale hierarchical image database," in *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition*, Miami, FL, USA, pp. 248–255, 2009.
[29] B. Deguerre, C. Chatelain and G. Gasso, "Fast object detection in compressed JPEG images," in *Proc. of IEEE Intelligent Transportation Systems Conf.*, Auckland, New Zealand, pp. 333–338, 2019.
[30] A. Hamayun, "Feature fusion and classifier ensemble technique for robust face recognition," *Signal Processing*, vol. 11, pp. 1–15, 2017.
[31] S. Liu, D. Huang and Y. Wang, "Learning spatial fusion for single-shot object detection," arXiv, 1911.09516, 2019.
[32] A. Jalal, A. Ahmed, A. A. Rafique and K. Kim, "Scene semantic recognition based on modified fuzzy c-mean and maximum entropy using object-to-object relations," *IEEE Access*, vol. 9, pp. 27758–27772, 2021.
[33] Z. Sun, G. Bebis and E. Miller, "Object detection using feature subset selection," *Pattern Recognition*, vol. 37, no. 11, pp. 2165–2176, 2004.
[34] A. Tariq, M. U. Akram and M. Y. Javed, "Lung nodule detection in CT images using neuro fuzzy classifier," in *Proc. of Fourth Int. Workshop on Computational Intelligence in Medical Imaging*, Singapore, pp. 49–53, 2013.
[35] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler *et al.,* "The cityscapes dataset for semantic urban scene understanding," in *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, Las Vegas, USA, pp. 3213–3223, 2016.
[36] N. Silberman, D. Hoiem, P. Kohli and R. Fergus, "Indoor segmentation and support inference from rgbd images," in *Proc. of European Conf. on Computer Vision*, Florence, Italy, pp. 746–760, 2012.

[37] N. Silberman, P. K. Derek Hoiem and R. Fergus, "Indoor segmentation and support inference from rgbd images," in *Proc. of European Conf. on Computer Vision*, Florence, Italy, pp. 746–760, 2012.
[38] M. Khodabandeh, A. Vahdat, M. Ranjbar and W. G. Macready, "A robust learning approach to domain adaptive object detection," in *Proc. of the IEEE/CVF Int. Conf. on Computer Vision*, Seol, South Korea, pp. 480–490, 2019.
[39] P. Wang, P. Chen, Y. Yuan, D. Liu, Z. Huang *et al.,* "Understanding convolution for semantic segmentation," in *Proc. of IEEE Winter Conf. on Applications of Computer Vision*, NV, USA, pp. 1451–1460, 2018.
[40] X. Song, C. Chen and S. Jiang, "RGB-D scene recognition with object-to-object relation," in *Proc. of the 25th ACM Int. Conf. on Multimedia*, New York, USA, pp. 600–608, 2017.
[41] X. Song, S. Jiang, L. Herranz and C. Chen, "Learning effective RGB-D representations for scene recognition," *IEEE Transactions on Image Processing*, vol. 28, no. 2, pp. 980–993, 2018.
[42] Z. Xiong, Y. Yuan and Q. Wang, "MSN: Modality separation networks for RGB-D scene recognition," *Neurocomputing*, vol. 373, pp. 81–89, 2020.
[43] D. Du, L. Wang, H. Wang, K. Zhao and G. Wu, "Translate-to-recognize networks for RGB-D scene recognition," in *Proc. of the IEEE/CVF Conf. on Computer Vision and Pattern Recognition*, Long Beach, CA, USA, pp. 11836–11845, 2019.