MSW MANAGEMENT -Multidisciplinary, Scientific Work and Management Journal
ISSN: 1053-7899
Vol. 36 Issue 1, 2026, Pages 6273-6279

VARIATIONAL LEARNING-DRIVEN TEXTURE SURFACE SEGMENTATION USING SVNET ARCHITECTURE

Prasanthi.Mandha[1],Research Scholar,Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation,Guntur(D), Andhra Pradesh. dhavalaprasanthi3@gmail.com
Dr.Vamsidhar Enireddy[2], Professor, Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation, Guntur(D), Andhra Pradesh. enireddy.vamsidhar@gmail.com

Abstract- Recently, various modeling techniques have been used to classify surface textures. Most methods utilize a lot of training data to analyze the surface texture, which raises the computational expenditure significantly. However, to categorize surface textures, a considerable number of training images is also needed for the application of different neural network algorithms. The suggested research offers a Segmentation with Variational Network Model (SVNet) for based on a deep learning framework. The method uses fewer training samples and uses a tailored parameter configuration to recognize surface texture. A segmentation with Variational Network Model (SVNet) is employed to categorize the produced surfaces with the proper labels into classes once the image descriptive vectors are created using numerical procedures to calculate the surface's pattern in reality. The suggested SVNet models are evaluated utilizing 16 texture classes from the Brodatz Image Textures dataset. The whole model is tailored to meet particular training needs, which gives suggested SVNet models a competitive edge over pre-trained networks. Additionally, the suggested SVNet method is contrasted with current methods to show the modern technique outcomes. The experimental results which achieve an accuracy of 96.46% for model outperform the traditional methods. Furthermore, the suggested SVNet models preserve a balance between computational expenses and accuracy.

Keywords- segmentation, deep learning, prediction, texture image, pre-trained network

1. Introduction

The field of surface texture evaluation in automated machine and image analysis is expanding quickly these days. In actuality, a wide range of applications require texture assessment. By offering information for identifying and interposing surfaces like wood, steel, sand, textiles, etc., the image parameters are utilized in this type of analysis to find defects and imperfections [1]. It entails taking features from the image surfaces, representing them, and then classifying them into the proper groups. In general, statistical techniques, structural methodologies, filter-related techniques, and model-related approaches are used to categorize and analyze surface texture recognition. Model-based techniques are frequently used in computer vision fields for surface texture identification and recognition. Additionally, when it came to identifying the surface texture of things, model-based methods such as auto-regressive systems [2], fractal designs, and supervised learning strategies like KNN, SVM, ANN, etc., outperformed the traditional methods [3]. For empirical and theoretical reasons, it is crucial to analyze and categorize the surface texture of an image taken in various lighting circumstances. Because the illumination pattern greatly influences the surface texture's visual properties, statistical learning is utilized in texture analysis to evaluate the objects' physical characteristics [4]. In addition to the statistically based methods, The author developed a method for material rotation and classification invariant to light source angle using photometric stereo. By employing statistically based machine learning techniques and transformed parametric designs, which have been covered in the suggested work, authors typically categorize the appearance of the surface acquired under various lighting sources. This will use fewer training image samples to achieve classification with significantly higher accuracy [5] – [6]. Is it possible to reduce the length of the training parameters in the actual image while still attaining a similar or higher categorization accuracy [7]. This is another main emphasis of the work that is being presented. Suppose this can reduce the computing cost of creating and analyzing the training samples. Similarly, bigger training sample sizes will result in more intricate model designs, further challenging surface texture research [8]. Additionally, authors have attempted to extract characteristics from input images with a size of 100 × 100 in the suggested study. Even with an extensive image dataset, the proposed technique can determine the characteristics faster [9] – [10]. The major research contributions are:

1) The proposed Segmentation with Variational Network Model (SVNet) model adopts variational function to attain segmentation with better segmentation accuracy. The model employs variational loss function to analyze the similarity among the output layers and feature functions.

2) Also, the minimization is attained using the proposed (SVNet) model with the Brodatz Image Textures dataset to acquire the enhanced segmentation performance.

This research is drafted as: section 2 demonstrates the wider analysis on various prevailing approaches. The methodology is explained in section 3 and its findings are explained in section 4. The work summary is given in section 5.

2. Related works

This section presents significant recent machine learning works for surface texture classification. When compared to other predictive modeling techniques, CNN's superior predictive accuracy is the primary factor contributing to its development as an algorithm using deep learning [11]. Additionally, this is primarily utilized in texture identification and image analysis activities. These techniques are used in various ways in the literature to determine the qualities of surface texture. The author [12] presented a method for categorizing textural characteristics in ultrasonic liver images. They employed a multi-resolution set of parameters to find the dispersed liver disorders. Images of estimated fractal dimensions obtained at different resolutions are gathered to create the feature vector, which includes the texture data. However, the Bayes Classifier and Hotelling trace criterion were used to determine the accuracy of classification for the features, which was found to be 90%. The author in [13] proposed a method for representing and identifying natural textures seen in various lighting and viewing scenarios using three-dimensional textons. They tested their approach on a sizable image collection made up of multiple materials and achieved an 87% rate of detection. A methodology that utilizes non-linear color modifications and lightening correction was proposed by [14] for facial feature detection under various lightening situations. They used a vast number of static color images, including both their own and images from other databases, to show the durability of the technique. To create a face, the machine learning system first finds the skin patches throughout the image. Next, the mouth, eyes, and border maps for the created face are created to confirm the results [15]. The author in [16] used Discrete Cosine Transformation (DCT) with minimum arithmetic processes to recognize and categorize edge elements in reduced samples. Nonetheless, cutting-edge results are obtained when the suggested methodology is tested on various still samples and an MPEG-I video frame. Additionally, the author in [17] showed how to use Multifractal Spectrum (MFS) to encode physical textures' geometrical and structural data at reduced dimensions. This study is a novel method for classifying surface texture using 3D geometric transformations since it is invariant to local affine and perspective lighting changes. Using neural networks, the author [18] developed a method for identifying existing cracks in digital images taken with a DSLR camera equipped with a macro LED light. The model demonstrated a 94% classification accuracy when tested against 11 samples with broken surfaces and varying illumination angles. The author in [19] developed a lane-detecting method using particle filter-based detectors. They created a system that uses images from railroad cameras to recognize train track lines. They used a shift in surface textures to identify lane edges. Filter-based surface texture evaluation provides superior outcomes compared to other lane identification methods. Researchers in [20] suggested a noise-robust adaptive hybrid pattern for noised texture evaluation. They put forth two models for characterizing textures, and the quantization algorithm is employed to lessen the sensitivity of the features to noise. Primitive micro features encode the global texture spatial structure for texture classification in their model [21]. Quantization is also used to determine the maximum entropy. Their findings demonstrate that this method can detect surface textures in three different kinds of noise. Furthermore, recent pattern identification and image analysis developments demonstrate extremely effective and encouraging outcomes. Using a trained convolutional neural network (CNN) to extract sweat pores from touchless, hidden, and touch-based fingerprint images, the author in [22] presented a new method for recognizing consistent fingerprint samples. The suggested approach correctly classified the majority of the fingerprints in the 23,000 classified pores in the dataset they utilized for their study. CNN was used by a researcher in [23] to check the railroad for flaws in the rail connectors and crossings. They used their framework to identify different textures, and the quick K-NN technique was used to compare the findings. They used two Local Binary Pattern (LBP) variants to extract characteristics. To recognize the broken structures, the author in [24] suggested a method for identifying the surfaces of ceramic tiles. A highly accurate automatic image analysis method is recommended to detect flaws.

They used the Rotation Invariant Measure of the Local Variance operator to detect edges on surfaces. Additionally, SVM is used to identify the corresponding ceramic features and name all the flaws. CNN was examined by [26] to analyze the material flaws. The aim of the work is to evaluate a CNN with random forests, Gabor filters, and PSO-ICA. Using CNN, they demonstrated that their outcomes are superior for 5 of the 7 categories of elements. The author in [26] used the CNN to identify stone tiles. Instead of employing CNN for the classification process, they employed the feature extraction approach as a CNN layer. They suggested an approach that uses a feature vector as input and moves the final CNN layer. To identify surface geometrical patterns, the authors in [27] proposed the edge-local binary sequence as a local descriptor. They used an irregular vertex dispersion to determine the surface texture for different objects. Their method reliably classifies things based on their surface structure and can recover three-dimensional patterns. The author in [28] used a patch-based approach that uses CNN to identify

MSW MANAGEMENT -Multidisciplinary, Scientific Work and Management Journal
ISSN: 1053-7899
Vol. 36 Issue 1, 2026, Pages 6273-6279

patches to segregate latent fingerprint images with complicated backgrounds. Using the IIIT-D latent fingerprint database, the suggested model's overall accuracy for fingerprint and non-fingerprint patched images is 83.90% and 94.44%, respectively.

The author in [29] have presented a novel method for surface texture classification using images from small datasets. The feature vectors were calculated using the Binary Dragon Fly Algorithm, and a unique DNN based on a fitness function was employed to optimize the system's precision. Additionally, the suggested approach analyzes the size of the training dataset using six different kinds of k-cross validations (where k takes values like 2, 3, 4, 5, 6, and 10). Regarding recognition accuracy, the suggested approach works better than the SVM strategy on two familiar datasets: KTH-TIPS and Textured Surfaces. A method for creating an innovative texture identifier utilizing local binary patterns and identifying the appearance with multi-kernel SVM was presented. The texture descriptor (IRSLBP) the suggested research produces is resilient against scale consistency, rotation, and illumination. Additionally, RGB quantization is used as a pre-processing technique to categorize the texture images accurately. The suggested method lowers computation time while maintaining a 95% classification accuracy [29]. By reducing geometric reprojection error, the author suggested a method for determining the framework of a surface of revolution. The Euclidean shape of the revolutionary surface is recreated using estimated shape and structure. Enough trials are conducted to assess their technique's accuracy and real-time performance. SVM was suggested as a method for categorizing flaws in images of the surfaces of steel strips. To categorize the surface textures, they acquired three different kinds of image data: shape, geometric, and grayscale features. SVM uses the Gauss radial function as its kernel function, and cross-validation is utilized to select the model parameters. The outcomes of their experiments indicate that surface classification's reliability is satisfactory compared to other traditional methods. An author in [30] presented a model to recognize variations in sample information on the surface. The spatial attraction technique is employed to identify modifications in pixels through improved MRF and the spatial pixels' neighborhood data, which improves the precision of spatial data in MRF and detects alterations in surface texture information. Some researchers have recently proposed classification and prediction based on different ML and computational modeling strategies [30].

3. Methodology

DL algorithms solve numerous difficulties requiring intricate, extremely variable functions. The methods generate characteristic networks using features from upper levels of the structure which are composed of several lower-level attributes, to address such issues. When employed to classify the surface characteristics in various domains, these characteristics yield high accuracy. The following section presents the suggested DL model and its sub-models for surface texture classification.

3.1. Dataset

Brodatz Image Textures are a classic and widely used dataset in the field of image processing, especially for **texture analysis**, **segmentation**, and **classification**. The characteristics are given below:

Grayscale Images: Typically 512×512 pixels, single-channel.

Texture Variety: Includes a wide range of **natural and artificial textures** such as:

- ✓ Fabrics (e.g., denim, canvas)
- ✓ Leather and animal skin
- ✓ Wood, brick, and stone
- ✓ Grass, sand, and foliage
- ✓ Paper and woven patterns

Uniform Illumination: Most images have consistent lighting, which simplifies certain segmentation and classification tasks.

No Labels: Original images do not include ground-truth segmentation or classification labels; researchers often create their own subsets.

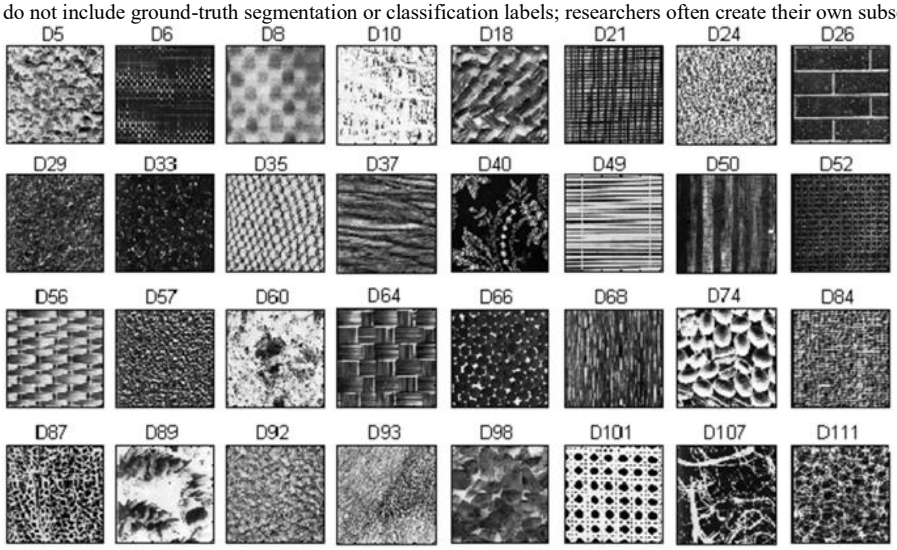


Fig 1. Dataset samples

3.2. Variational Model for texture image segmentation

Particularly, for an extracted image with dimension $u(r), r \in \Omega \subset R^2$, the variational functional model is employed for the smooth classification of images into N class piecewise consistent segments which is expressed in Eq.(1):

$$L_{MS}(\mathrm{X};\mathrm{u}) = \varepsilon(\mathrm{X},\mathrm{u}) + \lambda \sum_{n=1}^{N} \int_{\Omega} |\nabla \mathrm{X}_n(\mathrm{r})| dr \tag{1}$$

$$\varepsilon(\mathrm{X},\mathrm{u}) = \sum_{n=1}^{N} \int_{\Omega} |\mathrm{u}(\mathrm{r}) - \mathrm{c}_\mathrm{n}|^2 \, x_n(\mathrm{r}) dr \tag{2}$$

and $\chi_n(r)$ indicates the characteristic function of class n such that:

$$\sum_{n=1}^{N} X_n(\mathrm{r}) = 1; \quad \text{for all } \mathrm{r} \in \Omega \tag{3}$$

and c_n indicates the mean pixel score represented by:

$$c_n = \frac{\int_{\Omega} u\,(\mathrm{r}) \mathrm{X}_\mathrm{n}(\mathrm{r}) \mathrm{dr}}{\int_{\Omega} u\,(\mathrm{r}) \mathrm{dr}} \tag{4}$$

By specifying $u(r)$ and c_n as vectors of each channel value is expanding this energy function for multichannel images such as color images is simple. Pixels in the identical location of an image with intensity in homogeneities may differ so significantly that they can't be attributed to a unique stable value. To solve this, the piecewise steady-state assumption and suggested a local intensity clustering approach as follows:

MSW MANAGEMENT -Multidisciplinary, Scientific Work and Management Journal
ISSN: 1053-7899
Vol. 36 Issue 1, 2026, Pages 6273-6279

$$u\,(\mathrm{r}) = b(\mathrm{r})\sum_{n=1}^{N} c_n X_n(\mathrm{r}) \tag{5}$$

Where $b(r)$ represents a bias field. After that, the term is expressed in Eq. (6):

$$\varepsilon(\mathrm{X},\ \mathrm{u},\mathrm{b}) = \int\left(\sum_{\omega\in I}\int_{\Omega} K(\mathrm{r}-\mathrm{r}')||\mathrm{u}(\mathrm{r}') - \mathrm{b}(\mathrm{r})\,c_w||\hat{}2\right)X_w(\mathrm{r}')\mathrm{dr}' \tag{6}$$

Where, kernel function is represented by $K(r - r')$. The kernel function K uses a truncated Gaussian function. Since $b(r)$ is computed using the estimated through $\{c_n\}$ when reducing the energy function, the proposed technique facilitates image classification with grey-level inhomogeneity correction. The characteristic function χ_n prevents the variational model and its extension with the bias from being distinctive. The multiphase level-set approaches were suggested to render the cost function differentiable. A set $\Phi = [\varphi_1..\varphi_p]$ should be specifically considered, where $\varphi_i: \Omega \to R$ indicates a level function starting at image domain Ω to the actual value that reflects the level-set's altitude. The vector Heaviside function is also defined by Eq. (7):

$$H(\varphi) = [H(\emptyset_1)\ldots \mathrm{H}(\emptyset_\mathrm{p}) \tag{7}$$

where $H(\varphi)\ equals\ 1$, when φ greater than 0 or $H(\varphi)\ equals\ 0$, otherwise. With this concept, N different classes within Ω can be represented by $N := 2_p$. For instance, Lanrange's equations at the level functions for the segmentation of $N = 4$ levels with 2 level functions (p equals two and N equals $\{00, 01, 10, 11\}$) are as follows:

$$\frac{\partial\emptyset_1}{\partial t} = \delta_\epsilon(\emptyset_1)\left\{\lambda div\left(\frac{\nabla\emptyset_1}{|\nabla\emptyset_1}\right) - [(\mathrm{u}-\mathrm{c}_{11})^2 - (\mathrm{u}-\mathrm{c}_{01})^2]H(\emptyset_2) + [(\mathrm{u}-\mathrm{c}_{10})^2 - (\mathrm{u}-\mathrm{c}_{00})^2](1-H(\emptyset_2))\right\} \tag{8}$$

$$\frac{\partial\emptyset_2}{\partial t} = \delta_\epsilon(\emptyset_2)\left\{\lambda div\left(\frac{\nabla\emptyset_2}{|\nabla\emptyset_2}\right) - [(\mathrm{u}-\mathrm{c}_{11})^2 - (\mathrm{u}-\mathrm{c}_{01})^2]H(\emptyset_1) + [(\mathrm{u}-\mathrm{c}_{01})^2 - (\mathrm{u}-\mathrm{c}_{00})^2](1-H(\emptyset_1))\right\} \tag{9}$$

The differential value of the $H(t)$ is the source of the approximation to the $1D$ Dirac delta function, represented by δ. The multiphase level-set approach can effectively split geographically dispersed sections of the same class because of the multi-dimensional lifting character of the level functions.

3.3. Image segmentation

One of the key findings in this research is that the variational energy model can be directly reduced by employing the softmax layer as a differentiable estimation of the characteristic function. This contrasts with traditional multiphase level set techniques, which use a vector $H(t)$ of multiphase independent variables to approximate the distinct function and produce a differentiable energy function. In particular, a neural network's n^{th} channel softmax result is expressed as follows:

$$y_n(\mathrm{r}) = \frac{e^{z_n(\mathrm{r})}}{\sum_{i=1}^{N} e^{z_i(\mathrm{r})}};\ \mathrm{n} = 1,2,\ldots,\mathrm{N} \tag{10}$$

Here, $r \in \Omega\ and\ z_i(r)$ indicate the result of the system at r from the previous layer preceding the softmax. When the pixel score at r is in class n, the output value is around 1.

$$\sum_{n=1}^{N} y_n\ (\mathrm{r}) = 1;\ \ for\ all\ r \in \Omega \tag{11}$$

Additionally, it is simple to demonstrate that, which essentially the same as the characteristic function feature. This resemblance makes it clear that the result of the softmax can function as a unique variation of the characteristic function for class membership. Therefore, the proposed technique suggests the variational model which is modeled after SVNet as a loss function:

$$\mathrm{L}_{MS}(\ominus;\ \mathrm{x}) = \sum_{n=1}^{N}\int_{\Omega}|x(\mathrm{r}) - c_n|^2 y_n\ (\mathrm{r})dr + \lambda\sum_{n=1}^{N}\int_{\Omega}|\nabla y_n(\mathrm{r})|dr, \tag{12}$$

Where, $x(r)$ represents the source, $y_n(r) := y_n(r;\ \Theta)$ represents the result of softmax layer and

$$\mathrm{c}_\mathrm{n} = c_n(\theta) = \frac{\int_{\Omega}|\mathrm{x}(\mathrm{r})y_n(r;\,\theta)|dr,}{\int_{\Omega}|y_n((r;\,\theta))|dr,} \tag{13}$$

It is the mean pixel score of the class n, where Θ indicates the learnable system attributes. Consider that Eq. (13) is differentiable concerning Θ. As a result, during the training process, back-propagation is used to reduce the loss function. Another intriguing feature is that the loss function offers a self-supervised loss for classification. When label-based supervised loss is unavailable, this feature is helpful for unsupervised segmentation setups. Additionally, even in semantic labels, a novel loss function results in image classification based on the distribution of pixel values that can be enhanced.

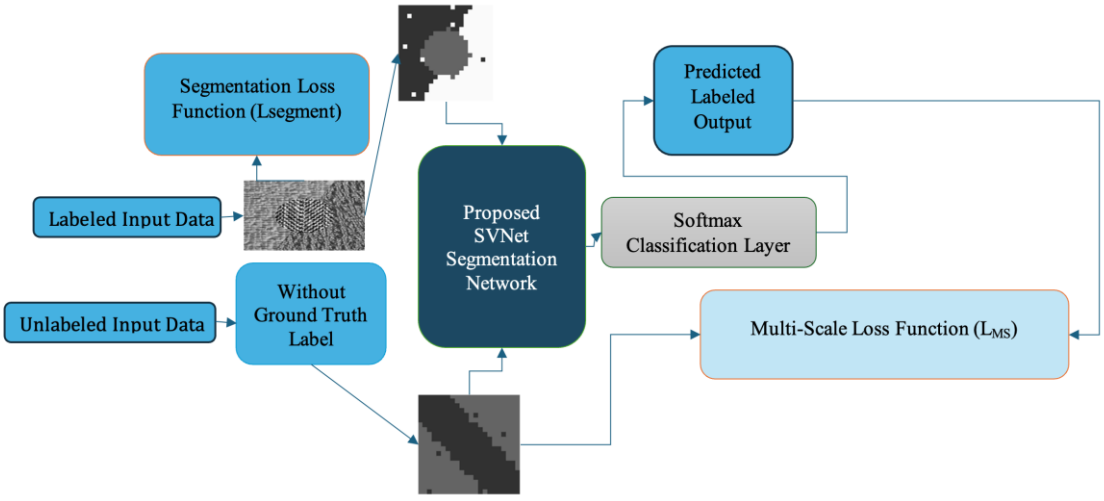


Fig 2. Image segmentation process

3.4. Cost analysis

One standard variational method for a given cost function is to resolve the appropriate Lagrange's:

$$-\lambda\left(\frac{\nabla \mathrm{y_n}}{|\nabla y_n|}\right) + \ (\mathrm{x}(\mathrm{r}) - \mathrm{c_n})^2 - \sum_{i\neq n}(\mathrm{x}(\mathrm{r}) - \mathrm{c_i})^2 = 0 \tag{14}$$

For every n, the restriction provides the final term. To solve the Lagrange's equation, a matching fixed point iteration is supplied by:

MSW MANAGEMENT -Multidisciplinary, Scientific Work and Management Journal
ISSN: 1053-7899
Vol. 36 Issue 1, 2026, Pages 6273-6279

$$y_n^{k+1} = y_n^k + \eta^k \left(\lambda \, div \left(\frac{\nabla y_n^k}{|\nabla y_n^k|} \right) + \sum_{i=1}^{N} (-1)^{\delta(n,i)} \left(x(r) - c_i^k \right)^2 \right) \quad (15)$$

Where $\delta(n, i)$ is a discrete Dirac delta function, η^k denotes a step length, and the superscript k indicates the k^{th} iteration. The proposed system can be understood as an unrolled iteration trimmed to a specific count of iterations, and is encouraged by the learned iterative soft-thresholding algorithm (LISTA). By contrasting it with the level set approach, this idea is demonstrated for unsupervised segmentation issues.

3.5. Loss analysis

Any supervised, semi-supervised, or unsupervised segmentation algorithm can be used with the new loss. In particular, an example application of the suggested loss function is shown in Fig 2. Both pixel-level annotated images X_L and unidentified images X_U are given as input into a DNN. The variational loss functional L_{MS} and the segmentation loss $Lsegment$ can be used to train the neural network for the images being provided X_L. By reducing L_{MS}, the system variables are revised for the unlabeled data X_U. In, unsupervised segmentation, B is the bias field which is utilized for the bias revised loss function of the variational model and is also estimated by the neural network tasks, respectively. Any segmentation network that creates segmentation maps from input images, whether or not they contain pixel-level annotated data, can be used as a DNN in this context. The inclusion of the variational functional L_{MS} is the only modification. Here, researchers provide greater detail on the variational loss's several applications. To train the DNN without any predicted or weak supervisory labels, the variational loss function is used for unlabeled data and the traditional classification loss function for pixel-wise labeled data if the segmentation dataset contains few pixel-level semantic annotations. The training loss function for the network is specifically made to employ both annotated and raw data in the way described below:

$$loss = \alpha L_{segment} + \beta L_{MS} \quad (16)$$

$$\alpha = \begin{cases} 1 & if\ image\ \ poses\ label \\ 0 & else \end{cases} \quad (17)$$

Here, β represents a hyper-parameter. $L_{segment}$ is generally the cross-entropy function expressed as:

$$L_{CE} = -\frac{1}{P} \sum_{i=1}^{P} \sum_{n=1}^{N} g_n(\mathrm{i}) \log y_n(\mathrm{i}) \quad (18)$$

Where, P represents the number of source pixels, $g_n(i)$ indicates the semantic label and $y_n(i)$ indicates the expected probability that pixel 1 belongs to the n^{th} class in the feature map. As a result, the suggested approach is trained solely by the loss function of variational when it receives input without pixel-wise labeling. Otherwise, the loss function of the variational model and the traditional classification loss are used to train the network to segment particular regions and categorize those regions. In the meantime, α is set to 1 during complete network training when fully pixel-wise annotation labels are present. However, the proposed neural network has been developed employing both $L_{segment}$ and L_{MS}, which allows it to consider pixel resemblance and semantic data to execute segmentation in greater depth, unlike the current supervised training approach that minimizes $L_{segment}$. Since this loss is self-supervised, by declaring α equal to zero, the cluster can also be estimated from the pixel statistics in the unsupervised image segmentation. The proposed functional model merely attempts to make each section have comparable pixel values using the contour size normalization. As a result, distinct segmented regions are frequently created if the image sample's pixel scores inside each section differ significantly because of intensity in homogeneities. While more semantic loss can lessen the issue, unsupervised learning does not allow this extra oversight. The above-discussed bias field estimation can offer more supervision details. To select $K(r - r') = \delta(r - r')$, researchers specifically apply the requirement that the bias field's total fluctuation be modest. It may thereafter be changed to:

$$y_n^{k+1} = y_n^k + \eta^k \left(\lambda \, div \left(\frac{\nabla y_n^k}{|\nabla y_n^k|} \right) + \sum_{i=1}^{N} (-1)^{\delta(n,i)} \left(x(r) - c_i^k \right)^2 \right) \quad (19)$$

Where $\gamma > 0$ indicates a constant parameter and

$$c_n = \frac{\int_\Omega b(r) x(r) y_n(r) dr,}{\int_\Omega b^2(r) y_n(r) dr,} \quad (20)$$

Which uses the input image $x(r)$ and the estimated bias to get the centroid c_n. As a result, by reducing, the NNs can determine the bias field and segmentation maps concurrently. The process of estimating the bias field in a neural network is simple to accomplish. In particular, the NN is built to provide an extra bias field $b_n(r)$ in addition to $\{z_n(r)\}$ that sends as input to the softmax layers. An identical NN can produce bias and bias-adjusted segmentation masks by enhancing the B channel before sending it to the softmax layers and reducing the revised loss function.

4. Experimental results

The backend used in the proposed framework is TensorFlow and the suggested SVNet framework is developed in Python using Keras repositories which are specially made for SVNets. This section analyzes and discusses the surface texture findings that were obtained. The dataset extracted from Brodatz Image Textures dataset was employed for the suggested study. The Canon digital camera and a Sigma 17 to 70-millimeter lens provided the training images. Surface texture classes, including textiles, rice, lentils, grass patches, and more, are included in the training dataset. To lessen the influence of bias in various textures, the dataset consists of images rotated by multiple angles, such as 30, 45, 60, 90, 220 degrees, etc., and images produced via patch normalization with an identical average gray value. Nevertheless, as indicated in Table 1, the training dataset comprises 1712 training samples with 16 texture classes. Fig 8 displays the images from the training dataset and outcomes. Two methods are used in the proposed study to train the SVNet algorithm. The first method uses a training dataset of 1712 image samples divided into 16 classes, comprising 343 test images and 1369 training images to assess the reliability of the suggested approach. The training samples are used to create the test images. The Model 1 takes about 1.5 hours to train on an Intel i5-7200U with 8 gigabytes of random access memory and GTX 940MX graphics, and it is trained on a batch length of thirty-two with fifty epochs. It shows the properties that the CNN learned from a source image of block length $100 * 100 * 1$ using a kernel dimension of 3 * 3 and thirty-two filters. To assess the efficiency of the SVNet model with fewer samples, Model 2 splits the first training dataset into two. The dataset used to test model consists of eight hundred twenty-four samples with seven texture classes, of which 165 are test images and 659 are training images. The suggested model is trained on the same system with a batch length of 32 and 35 epochs, taking about 0.45 hours.The algorithm's features using the same image block length and kernel are displayed. For the two training datasets, the precision and loss curves are produced. The accuracy and loss curves for the suggested model are displayed in Fig 3 and Fig 4, respectively. The loss curve shows a graph of loss vs. epochs. Fig 4 demonstrates that the loss function gradually reduces as the amount of epochs increases. The training and validation loss curves were first identified at their highest 2.75 at epochs = 0. At epochs = 8, the loss curves then drop almost to 0.7. At epochs = 22, the validation loss exhibits signs of over-fitting and increases to 0.75, while the training loss starts to decrease and attains approximately 0.0024. Nevertheless, the validation loss curve drops to roughly 0.40 between epochs 23 and 34. The training loss remains at 0.0024 at epoch = 35, whereas the validation loss increases marginally to around 0.5. Ultimately, it is discovered that the loss produced by the suggested approach for 50 epochs is 0.0018 for training images and 0.2143 for test images. However, the accuracy curves visually depict the relationship between the model's accuracy and the training epochs. Fig 3 illustrates how the model's accuracy rises as the amount of epochs grows. The validation and training accuracy rate was initially determined to be 0.05 on epoch 0. It was then discovered that the training and verification accuracy calculated at epoch = 7 reached roughly 0.8. Nonetheless, it was noted that the training accuracy increased steadily from epoch = 9 to 18 and went around 1.0 (100%), while the mean validation accuracy was identified as 0.83. The accuracy decreases to 0.77 and 0.785 for epochs 21 and 34, respectively. Ultimately, the validation accuracy is determined to be 1, and the training accuracy is determined to be 0.9243 at epoch = 50. However, because of the nature of the dataset, researchers can overlook this over-fitting issue in real-world scenarios. Similarly, Fig 3 and Fig 4 display the accuracy and loss curves produced by the suggested model after it has been trained on a dataset of eight hundred twenty-four samples with seven classes. Nevertheless, at epoch = 35, the error calculation value produced by the suggested framework is determined to be 0.2109 for training images and 0.0741 for testing samples. Rapid variations in accuracy were seen in specific locations during epochs 6–13. For example, this accuracy level increases to approximately 0.85 at epoch = 8 and decreases to approximately 0.59 at epoch = 9. Additionally, it is essential to note that the model shows less over-fitting. As a result, it demonstrates that the suggested model learns strong features and generalizes well with few training images.

4.1. Classification results

Several categorization measurement strategies are used to assess the performance of the suggested models. The output is shown in Table 1 according to several performance measures that assess the models' and states' accuracy. Below is a discussion of these performance measurement parameters:

4.2. Confusion matrix

The confusion matrix assesses the model's classification precision by sketching a matrix that displays the model's accurate and inaccurate predictions and counts. It only functions properly if all of the classes' data are standardized. The confusion matrix plot generated by the suggested categorization models is displayed in Fig 8. However, to assess the confusion matrix, authors must comprehend its 4 features:

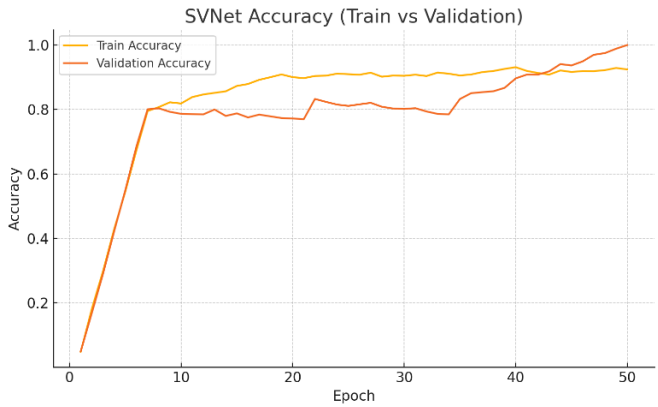
MSW MANAGEMENT -Multidisciplinary, Scientific Work and Management Journal
ISSN: 1053-7899
Vol. 36 Issue 1, 2026, Pages 6273-6279

i) True Positives (TP): This happens when the original class value matches the expected class value and is recognized to be accurate.
ii) True Negatives (TN): These suggest that the real class value agrees with the expected class value and recognizes it as inaccurate.
iii) False Positives (FP) and False Negatives (FN): When the real class and the predicted class differ, or the converse occurs, these values are known as FP and FN.

4.3. Classification Accuracy

The framework's precision in classification is determined by averaging the scores on the confusion matrix's principal diagonal, which is determined by Eq. (21) to Eq. (24). *N* in the equation represents the dataset's number of testing images. The numbers at the bottom of the confusion matrix in Fig 8 are called predicted scores, whereas the ones at the left are called actual ones. The suggested model performed effectively across 343 testing samples, as evidenced by the confusion matrix plot in Fig 8, where the anticipated value for most classes matched the actual values. For example, it was discovered that the real and predicted values for class "blanket2" were accurate and equal to 32. However, the system's reliability decreased in certain instances due to the model's poor prediction performance. For example, the model correctly classified 22 cases of the class "rug1" but incorrectly classified 10 instances of the same class. Likewise, Fig 8 shows the confusion matrix plot that reveals the suggested model identified accurate predictions on 165 training images.

Fig 3. Accuracy comparison



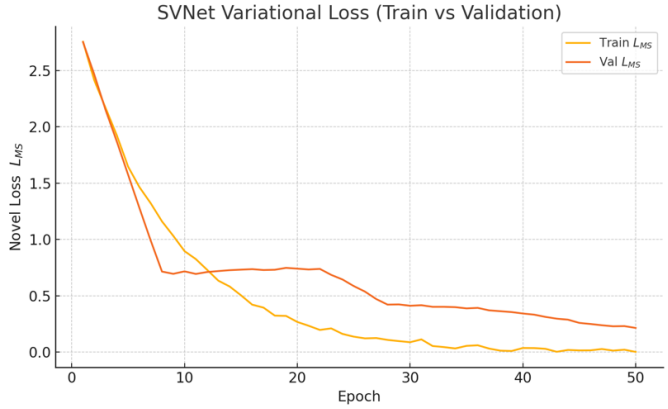
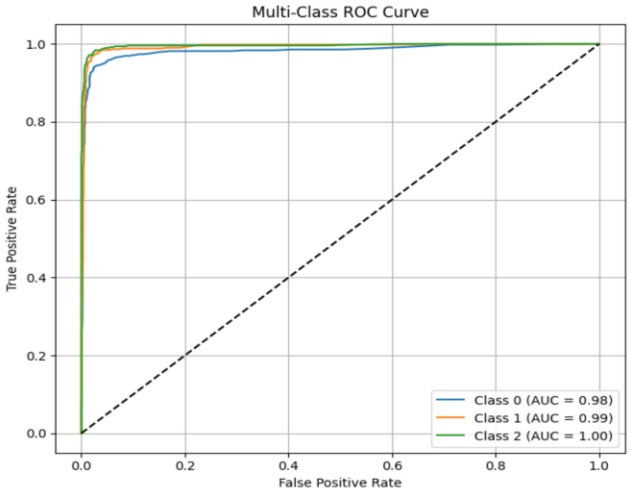


Fig 4. Loss comparison



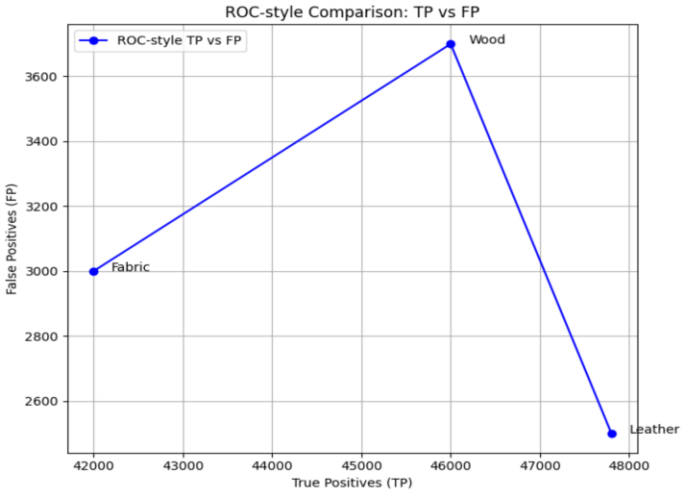


Fig 5. ROC comparison along with multi-Class

Test data Report :

	precision	recall	f1-score	support
Fabric	0.93	0.91	0.92	46000
Wood	0.93	0.94	0.93	49000
Leather	0.95	0.96	0.95	50000
accuracy			0.94	145000
macro avg	0.94	0.94	0.94	145000
weighted avg	0.94	0.94	0.94	145000

Fig 6. Testing outcomes

Precision: It is a ratio of accurately estimated positive findings to all the positive findings that the classification framework predicted. Eq. (21) is used to calculate this.FP denotes the model's projected false positive value. However, there will be less inconsistency between the actual and anticipated output if the accuracy value is higher and the false positive rate is lower.Recall (R): Its value should be higher than 0.5 and represent the proportion of accurately positive outcomes to all pertinent samples that the network model determined to be positive. The recall for the suggested model is determined using Eq. (22).
where FN is the classifier's projected false negative value.
F1-score: This parameter provides the calculated average of precision and recall. It shows how accurate and reliable the trained classifier is. The closed interval [0, 1] contains the F1-Score value. The F1-Score is represented mathematically by Eq. (21) to Eq. (22).

$$Precision = \frac{TP}{TP + FP} \tag{21}$$

$$Recall = \frac{TP}{TP + FN} \tag{22}$$

$$F1 - score = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{23}$$

$$MIoU = \frac{1}{k+1} \sum_{i=0}^{k} \frac{TP}{FN + FP + TP} \tag{24}$$

MSW MANAGEMENT -Multidisciplinary, Scientific Work and Management Journal
ISSN: 1053-7899
Vol. 36 Issue 1, 2026, Pages 6273-6279

Table 1. Segmentation outcomes

Types	Precision (%)	Recall (%)	F1-score (%)	MIoU (%)	p-value
YOLO	90	91	90	90	1.5
SOLO	89	90	91	91	1.8
RCNN	90	82	83	84	2.6
UNet	85	86	86	87	3.5
Swin transformer	87	87	87	88	1.1
SETR	86	88	85	86	1.8
SVNet	**92**	**92**	**92**	**93**	**0.18**

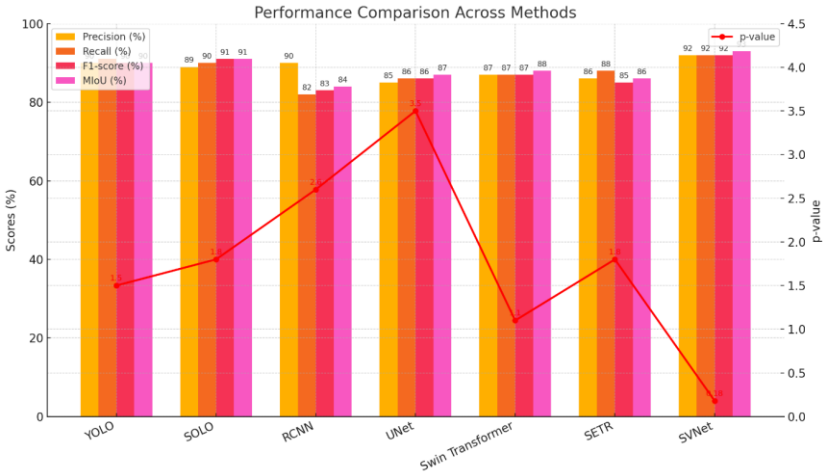


Figure 7. Comparison Graphs on Segmentation outcomes with various approaches

Table 1 and Fig 7 shows the comparison of proposed vs. existing approaches with metrics like precision, recall F1-score, MIoU and p-value. The comparison is made among YOLO, SOLO, RCNN, UNet, Swin transformer, and SETR respectively. The proposed MS-DCNM gives better results compared to the existing approaches. The proposed SVNet model gives 92% for precision, recall and F-score, 93% for MIoU and 0.18 for p-value respectively.

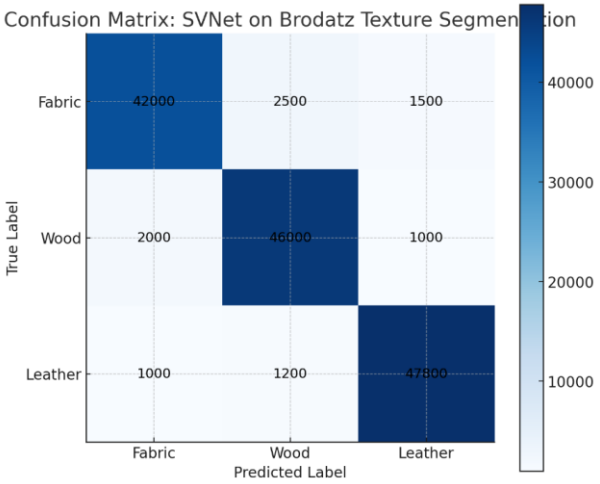


Fig 8. Confusion matrix

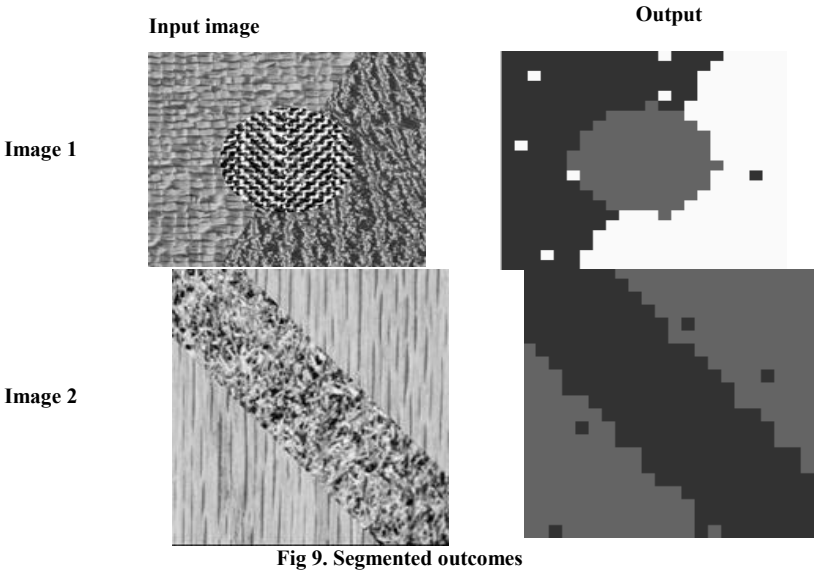


Fig 9. Segmented outcomes

MSW MANAGEMENT -Multidisciplinary, Scientific Work and Management Journal
ISSN: 1053-7899
Vol. 36 Issue 1, 2026, Pages 6273-6279

4.4. Discussion

The outcomes of the suggested SVNet models are shown in Table 1. It indicates, however, that the result of the segmentation for the suggested training model assessed with fewer samples yielded higher precision than the suggested training model with 1712 samples. Additionally, a seeding function is implemented at the start of SVNet algorithm to guarantee that the developed models produce the most accurate findings and avoid the randomness issue that most SVNet algorithms encounter. As a result, the suggested models' efficiency increases, and their operating costs are often reduced. Additionally, the findings are contrasted with the cutting-edge texture recognition techniques to show the strength of the suggested method. Table 1 presents a comparative examination of the segmentation accuracy performance of different methodologies. The studies in Table 1 are related to surface texture identification, where the authors employed different datasets. The Fig 9 shows the segmented outcomes over the input images. However, the primary objective of evaluating the research is to demonstrate the accuracy of surface appearance feature classification. The findings showed that the suggested approach outperforms the current methods concerning texture feature classification.

5. Conclusion

This study uses pattern recognition theory to categorize surface topography for various engineering uses. A unique method for classifying surface texture into different classes based on Segmentation with Variational Network Model (SVNet) is proposed. Two distinct models are used to train the suggested SVNet. Offering tailored training criteria further improves the benefit over previously trained neural network models. However, 92% segmentation accuracy is attained when the suggested model with one thousand twelve samples is evaluated on the training dataset. However, the model performs better when the training dataset is reduced, with an accuracy of 92%. The dataset from Brodatz Image Textures dataset includes classes of leather, glass, paper etc., as well as patches employing different pre-processing approaches, which are utilized to assess the suggested method. Several performance measures are used to gauge the outcomes. The functionality of the proposed design demonstrates that the accuracy of surface texture detection is superior to that of other traditional methods. This aids in the pattern identification community's scientific identification of different surface features. However, the suggested approaches do not have the drawback of utilizing labeled data, which is required for supervised learning. To continue this work, researchers typically use the suggested models in semi-supervised or unsupervised learning, which entails training using unlabeled data.

REFERENCES

[1] Zhu Z, You X, ChenCP, Tao D, Ou W, Jiang X, Zou J (2015) An adaptive hybrid pattern for noise-robust texture analysis. Pattern Recognition 48(2015):2592–2608
[2] ZhangZ, HongW-C, LiJ(2020) Electric load forecasting by hybrid self-recurrent support vector regression model with Variational mode decomposition and improved cuckoo search algorithm. IEEE Access 8: 14642–14658
[3] ZhangZ,HongW-C(2019) Electric load forecasting by complete ensemble empirical mode decomposition adaptive noise and support vector regression with quantum-based dragonfly algorithm. Nonlinear Dynamics 98(4):1107–1136
[4] YongHua X, Cong WJ (2015) Study on the identification of the wood surface defects based on texture features. Optik- International Journal for Light and Electron Optics 126(19):2231–2235
[5] Veerashetty S, Patil NB (2019) Novel LBP based texture descriptor for rotation, illumination and scale invariance for image texture analysis and classification using multi-kernel SVM. Multimed Tools Appl:1 21
[6] Tao X, Zhang D, Ma W, Liu X, Xu D (2018) Automatic metallic surface defect detection and recognition with convolutional neural networks. Applied Sciences, vol. 8, no. 9
[7] ThompsonEM,Biasotti S (2018)Description andretrieval of geometric patterns on surface meshes using an edge-base d LBP approach. Pattern Recogn 82(2018):1–15
[8] ParkJ-K, KwonB-K,ParkJ-H, KangD-J(2016) Machine learning-based imaging system for surface defect inspection. International Journal of Precision Engineering and Manufacturing-Green Technology 3(3):303 310
[9] Liu C, Hu W (2019) Real-time geometric fitting and pose estimation for surface of revolution. Pattern Recogn 85(2019):90–108
[10] Labati RD, Genovese A, Muñoz E, Piuri V, Scotti F (2017) A novel pore extraction method for heterogeneous fingerprint images using convolutional neural networks. Pattern Recogn Lett 113:58–66
[11] Kumar M, Srivastava S (2018) Image authentication by assessing manipulations using illumination. Multimed Tools Appl 78(9):12451–11246
[12] Khan AI, Wani MA (2018) Patch-based segmentation of latent fingerprint images using convolutional neural network. Appl Artif Intell, pp. 1–15
[13] Huang Y, Wang Y, Tai Y, Liu X, Shen P, Li S, Li J, Huang F (2020) Curricular face: adaptive curriculum learning loss for deep face recognition. In proceedings of the IEEE/CVF conference on computer vision and pattern recognition (pp. 5901-5910)
[14] Hanzaei SH, Afshar A, Barazandeh F (2017) Automatic detection and classification of the ceramic tiles' surface defects. Pattern Recogn 66(2017):174–189
[15] Goyal V, Singh G, Tiwari O, Punia SK, Kumar M (2019) Intelligent skin Cancer detection Mobile application using convolution neural network. Journal of Advanced Research in Dynamical and Control Systems 11(7):253–259
[16] Gu W, Lv Z, Hao M (2017) Change detection method for remote sensing images based on an improved Markov random field. Multimedia Tools and Applications volume 76, pages17719–17734(2017), vol. 76, pp. 17719–17734
[17] Chatra K, Kuppili V, Edla DR (2019) Texture image classification using deep neural network and binary dragon Fly optimization with a novel fitness function. Wirel Pers Commun 108(3):1513–1528
[18] Dong Y, Zhang Z, Hong W-C (2018) A Hybrid Seasonal Mechanism with a Chaotic Cuckoo Search Algorithm with a Support Vector Regression Model for Electric Load Forecasting. Energies 11(4):1009
[19] Cho M, Kim T, Kim IJ, Lee S (2020) Relational deep feature learning for heterogeneous face recognition. arXiv preprint arXiv:2003.00697
[20] Chatra K, Kuppili V, Edla DR (2019) Texture image classification using deep neural network and binary dragon Fly optimization with a novel fitness function. Wirel Pers Commun 108(3):1513–1528
[21] 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 Trans. Pattern Anal. Mach. Intell., vol. 40, no. 4, pp. 834–848, 2018.
[22] Qi, Z. Liu, J. Shi, H. Zhao, and J. Jia, "Augmented feedback in semantic segmentation under image level supervision," in Proc. Eur. Conf. Comput. Vis. Springer, 2016, pp. 90–105.
[23] Dhanachandra, K. Manglem, and Y. J. Chanu, "Image segmentation using K-means clustering algorithm and subtractive clustering algo rithm," Procedia Computer Science, vol. 54, pp. 764–771, 2015.
[24] Beck, T. Duong, H. Azzag, and M. Lebbah, "Distributed mean shift clustering with approximate nearest neighbours," in 2016 Int. Joint Conf. Neural Netw. (IJCNN). IEEE, 2016, pp. 3110–3115.
[25] Tran, "A fully convolutional neural network for cardiac segmentation in short-axis MRI," arXiv preprint arXiv: 1604.00494, 2016.
[26] Hung, Y.-H. Tsai, Y.-T. Liou, Y.-Y. Lin, and M.-H. Yang, "Adversarial learning for semi-supervised semantic segmentation," in Proc. British Mach. Vis. Conf. (BMVC), 2018.
[27] Pham, T.-T. Do, N. S¨ underhauf, and I. Reid, "SceneCut: Joint geometric and object segmentation for indoor scenes," in 2018 IEEE Int. Conf. Robotics and Automation (ICRA). IEEE, 2018, pp. 1–9.
[28] Le, K. G. Quach, K. Luu, C. N. Duong, and M. Savvides, "Reformulating level sets as deep recurrent neural network approach to semantic segmentation," IEEE Trans. Image Process., vol. 27, no. 5, pp. 2393–2407, 2018.
[29] Kanezaki, "Unsupervised image segmentation by back-propagation," in 2018 IEEE Int. Conf. Acoust. Speech. Signal Process. (ICASSP). IEEE, 2018, pp. 1543–1547.
[30] Kim and J. C. Ye, "Cycle-consistent adversarial network with polyphase U-Nets for liver lesion segmentation," 2018.