Applied Mathematics & Information Sciences
An International Journal

http://dx.doi.org/10.18576/amis/120216

A Framework for Labeling Images through Object Detection and Segmentation Using Preprocessing and ReNet Architecture

N. Shanmugapriya[1,*] *and D. Chitra*[2]

[1] Department of IT, Oxford Engineering College, Trichy, Tamilnadu, India.
[2] Department of Computer Science and Engineering, P.A College of Engineering and Technology, Pollachi, Tamilnadu, India.

Received: 23 Jan. 2018, Revised: 23 Feb. 2018, Accepted: 27 Feb. 2018
Published online: 1 Mar. 2018

Abstract: In a Segmentation-based approach, an image is segmented and its various regions are classified, unlike classifying the individual pixels. This papers uses the ReNet architecture to extract the features of an object in an image. This ReNet architecture replaces each convolutional layer(CNN) with four RNNs that also brings together lower-layer features from different directions. After the extraction of feature the image is over segmented into superpixels first and then it is classified into individual superpixels. The dependencies to the nearby superpixel labels shall be explored and exploited by Conditional Random Field statistical approach. Though the time to segment and label the images is somewhat higher, the pixel accuracy is more when this technique is implemented in the two datasets SIFT Flow and Stanford Background Dataset.

Keywords: Fully Convolution Networks (FCN), ReNet, Superpixel, mCRF, Full Scene labeling

1 Introduction

Scene labeling involves the scene understanding task which assigns a label to each pixel. Full scene labeling (FSL) or scene parsing is a process that involves labeling of each pixel in a given scene with the category to which the object belongs. The dependency of the pixel category may include either relatively short-range information or very long-range information or both.

Labeling demands contextual information because the labels tend to be dependent across pixels. Further, every image consists information that is required to label pixels at several levels. Convolutional neural networks or ConvNets are special kind of neural networks which attempts to reduce the number of parameters required to process large images by exploiting the locality of data in images.

Convolutional Neural Networks (CNNs) are implemented in various image recognition tasks including image classification and object detection and have been proved to be successful to a great extent. Classification CNNs are smoothly transformed into Fully Convolutional Networks (FCNs) by replacing fully-connected layers with 1x1 Convolutional layers and this process involves taking an image of arbitrary size of an object and calculating a semantic label map.

Although FCNs have provided almost perfect results in semantic segmentation, it also is subject to certain limitations, especially in modeling the distant contextual regions. But the crucial factor is that these distant contextual regions play vital role in reasoning and predicting contextual evidences in semantic segmentation. Moreover, the receptive field of a neuron in the Convolutional layer of FCNs generally corresponds to a local area of an input image. For example, when the middle area of an image is labeled, looking at the patterns of the sea on top and a hill at the bottom of an image leads to the exact prediction of the image of a 'beach,' but the limited size of the local receptive fields hinders the FCN in capturing such long-range dependence across various local areas. Though the receptive field could be adjusted to cover the whole image, the percentage of success is only limited in the process of encoding long-range context [1].

While ConvNet deals with only local information, ReNet [2] spans across the whole image through

* Corresponding author e-mail: shanmugapriyaphd@outlook.com

negotiating with the lateral connections. These lateral connections remove or resolve redundant features present throughout the image and attempt to realize a more compact feature representation of a given input image at each and every layer. ReNet also allows small displacement of features across multiple consecutive patches. In order to improve the segmentation some pre-processing methods are implemented. One of the widely used method is superpixel to segment the image.

The superpixels [3] provide compact and perceptual meaning to atomic regions of images. Each and every pixel in a superpixel indicates the meaningful atomic regions of the image of same object. While brought together, these pixels form the original object and the interaction between different objects, which is generally hard, becomes easier with the help of Superpixel. For Classifiers to get the accurate segmentation, Multiscale Conditional Random Field is used. mCRF [4] framework comprises three separate components, operating at three different scales: a local classifier, regional features, and global features.

In this context, three approaches are proposed in this paper to produce the final labeling with improved accuracy and better visual coherence: (i) maximizing the overall likelihood that each segment will contain a single object using the ReNet architecture, (ii) further classification using superpixel assigning single class to each of the superpixel, and (iii) applying multiscale conditional random field over a set of superpixels for accurate pixel classification through model joint probabilities.

2 Related Work

An approach in Recurrent convolutional neural network was proposed [5] that allows a large input context whereas the capacity of the model is limited. This method completely relies on recurrent architecture for convolutional neural networks, in which a sequential series of networks share the same set of parameters. There are various advantages in this method including (i) non-requirement of engineered features, as deep learning architectures are used to train the discriminative filters effectively in an end-to-end mode, (ii) the phase in which prediction is done does not depend on label space searching.

A related architecture [6] was introduced with eight learned layers in which five convolutional and three fully-connected layers of the object are found. For faster training, non-saturating neurons and an effective GPU implementation are used in this method. In this method only supervised pre-training is done to obtain required computational power to increase the size of the network.

Combining local classifiers with probabilistic models of label relationships was presented in problem of object detection [4], which is a more general task of image labeling. A basic difference between these existing earlier models and this proposed model is the form of the representation over labels. Capturing of label relationships through a more conceptual graphical method is one such model, which includes abstraction hierarchy that consists scenes, objects, and features. The distribution over labels shall be obtained based on pairwise relationships between labels at different sites.

A new method for Full Scene Labeling or Scene Parsing [7] was used in a Multiscale Convolutional Network to extract dense feature vectors and a tree of segments is computed from a graph of pixel dissimilarities. In this method each node is encoded by a spatial grid and a classifier is applied to produce a histogram that measures the impurity of the segment. The pixels are then individually labeled by a minimally-impure node over it, a segment that best explains the pixel's class.

In a hierarchical segmentation tree, a technique [8] was implemented to represent the image so that the resulting energy combining unary and boundary terms can still be optimized using graph cut (with all the corresponding benefits of global optimality and efficiency).

Another approach [3] was considered to detect the object as a multi-label superpixel method for labeling problem by minimizing an energy function. The data cost term is used to capture appearance, smooth cost term for encoding the spatial context and label cost term to favor compact detection. The data cost is thus learned through a convolutional neural network and the related parameters in the labeling model are learned through a structural SVM.

The use of deep learning techniques [9] was identified to deal with scene labeling, where off-the-shelf features of segments are recursively merged to assign a semantic category label. In contrast, this technique uses the ReNet architecture to parse the scene with a smoother class annotation.

A deep learning strategy [10] was used for scene parsing, i.e. to assign a class label to each pixel of an image. This approach uses the deep convolutional network for modeling the complex scene label structures, relying on a supervised greedy learning strategy. This strategy does not need hand crafted features. This is the advantage over CRF method. Another two advantages are (i) its inference does not involve searching the label space but simply requires the forward evaluation of a function, and (ii) discriminative training is performed efficiently through Stochastic Gradient Descent (SGD), without the need for estimating any normalization factor.

A hierarchical random field model [11] was introduced, which allows integration of features computed at different levels of the quantisation hierarchy. MAP inference in this model can be performed efficiently using powerful graph cut based move making algorithms. This approach proposed a novel hierarchical CRF formulation of object class segmentation that allows the quantisations of image space by unifying multiple

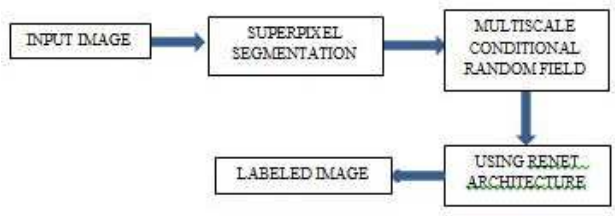

Fig. 1: System Architecture.

disparate, thus avoiding the necessity to make an appropriate decision.

A hierarchical approach [12] was proposed for labeling semantic objects and regions in scenes. This approach used a decomposition of the image in order to encode relational and spatial information. It bypassed a global probabilistic model and instead directly trained a hierarchical inference procedure inspired by the message passing mechanics of some approximate inference procedures in graphical models.

3 System Description

Scene Labeling system extracts relevant contextual information from raw pixels by combining the following preprocessing approaches: **??** ReNet Architecture so as to capture the global contexts as well as exhibiting its property of efficient parallelization, **??** Superpixel Segmentation, **??** Multiscale Conditional Random Field is defined with reference to a set of superpixels. Here using the ReNet architecture, features are extracted and simultaneously superpixels are segmented and consequently applying the Conditional Random Field the pixels are classified accurately and later labeled. The Architecture diagram of the proposed system is given in Fig. 1.

3.1 ReNET Architecture

The Network architectures have various properties which make them an optimal choice for sequence labeling: their flexibility in their use of context information such as they decided what to store and what not to store; their flexibility in accepting various types of data and their representations; and their flexibility in recognizing sequential patterns in the presence of sequential distortions.

The significant parameters those define the architecture of the ReNet include, the number of ReNet layers (N_{RE}), their corresponding receptive field sizes $(w_p \times h_p)$ and feature dimensionality (d_{RE}), the number of fully-connected layers (N_{FC}) and their corresponding numbers (d_{FC}) and types (f_{FC}) of hidden units [2].

When the CNN proves to be successful, especially, in computer vision, Recurrent Neural Networks (RNN) have been chosen by many in order to model sequential data, such as text and sound. The recurrent layers relatively consider the totality of the image while extracting the features of the specific location within the whole image. Whereas the CNN considers only the local context window in the process of extracting the feature of the image [2].

In the process of RNN, the lowest layer of the model moves over the input image, and in the same manner the subsequent layers operate on extracted representations from the layer below to form a hierarchical representation of the input.

3.1.1 Comparison of Renet and Convolutional Neural Networks

At each and every layer, both the ReNet and CNN apply the same set of filters to all patches of the input image or of the features those map with lower layer [2]. ReNet retrieves the information through lateral connections that cover the entire image, whereas the CNN uses only local information from the image. The lateral connections must help in extracting a more compact feature representation of the input image in each and every layer, which shall be achieved by the lateral connections either by removing or resolving repeated features at various locations of the whole image. This allows ReNet to resolve small displacements of features through multiple consecutive patches.

Max-pooling, which is used in CNN, proves to be problematic when building a convolutional auto encoder whose decoder is an inverse of CNN, because the max operator in CNN is not invertible. The ReNet is an end-to-end network with smooth and differentiable features, thus making it suitable for using it as a decoder in the auto encoder or in any of its probabilistic variants [7].

Notwithstanding the above, each layer of the ReNet is considered as an alternative of convolution + pooling layer, in which the pooling layer is replaced by the lateral connections, and as a result the convolution is done without any overlapping. Similarly, another variant of the usual CNN without any pooling also exist in an approach which uses convolution with a larger step to compensate the lack of reduction of dimension by pooling at each layer. However, this approach is different from the ReNet as each feature activation at a layer is done with reference to a subset of the input image but not with the whole input image.

3.1.2 Disadvantage of ReNET

The main disadvantage of ReNet is that it does not easily parallelize, because of the sequential nature of the recurrent neural network (RNN). On the other hand, CNN is highly parallelizable because its computing activation at each layer is highly independent [2].

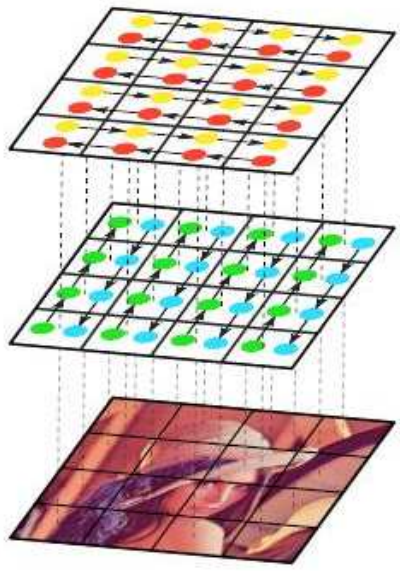

Fig. 2: A One Layer RENET.

3.1.3 Architecture of ReNet

In the ReNet architecture as shown in Fig. 1, $X = \{x_{i,j}\}$ denotes the input image or the feature map from the lower layer, where $X \in R^w \times h \times c$ with w indicates width, h indicates height, and c indicates the number of channels or the feature dimensionality, respectively [2]. Considering the receptive field (or patch) size of $w_p \times h_p$, the input image X is split into a set of $I \times J$, which is non-overlapping, patches $P = \{p_{i,j}\}$, where $I = w/w_p$, $J = h/h_p$ and $p_{i,j} \in R^{w_p \times h_p \times c}$ are the (i,j)th patch of the input image. The horizontal index is i and j is the vertical index.

First, the image is vertically swept with the two RNNs, where one RNN is working in a bottom-up approach and the other is working in a top-down apporach. Each RNN is taken as an input one patch at a time which is flattened and updates its hidden state, working along each column j of the split input image X [2].

$$V^F_{i,j} = f_{VFWD}(Z^F_{i,j-1},\ p_{i,j}), \quad \text{for} \quad j = 1,\ldots,J \quad (1)$$

$$V^R_{i,j} = f_{VREV}(Z^R_{i,j+1},\ p_{i,j}), \quad \text{for} \quad j = J,\ldots,1. \quad (2)$$

After this vertical, bidirectional sweep, combining of the intermediate hidden states $v^F_{i,j}$ and $v^R_{i,j}$ at each location (i,j) is done to get a composite feature map $V = \{v_{i,j}\}$, $j = 1,\ldots,J$, $i = 1,\ldots,i$, where $v_{i,j} \in R^{2d}$ and d are the numbers of recurrent units. Each $v_{i,j}$ now indicates the activation of a feature detector at the location (i,j) with regard to all the patches in the jth column of the original input ($p_{i,j}$ for all i) [2].

Thus obtained feature map V is swept over horizontally with two RNNs (f_{HFWD} and f_{HREV}). In a similar way as it was done with the vertical sweep, these RNNs work along each row of V producing the output feature map $H = \{h_{i,j}\}$, where $h_{i,j} \in R^{2d}$. Now, each vector $h_{i,j}$ symbolizes and represents the features of the original image patch $p_{i,j}$ in the context of the whole image.

In Fig. 2 the blue and green dots on the input image/feature map represent the steps of $f^{\downarrow}$ and $f^{\uparrow}$ respectively. By concatenating the resulting feature maps, the $f^{\rightarrow}$ (yellow dots) and $f^{\leftarrow}$ (red dots) are subsequently swept. Finally the resulting feature maps are concatenated in order to produce the output of ReNet layer which is depicted as a blue heatmap in the Fig. 2.

Here φ denotes the image map function of the input image X to the output feature map H as stated in Fig. 2. Clearly, the multiple φ's are stacked to make the ReNet deeper in order to capture increasingly complex features of the input image. After applying a number of recurrent layers to an input image, the activation at the last recurrent layer may be flattened and fed into a differentiable classifier.

3.2 Superpixel Segmentation

The ideal superpixel partition for detection depends on the minimum number of superpixels so as to increase the efficiency in inference so that each superpixel does not span in multiple objects [3]. Pre-segmentation using superpixels extracts features and categories from each and every segment and also from other combinations of neighboring segments [7]. Predicting the features and categories of each pixel independently from the neighboring segments leads to noisy predictions [7]. Hence a simple cleaning up is required by exploiting local regions of same color intensities that are assigned a single label. Then classifying each image by location, aggregating these predictions in each superpixel, and computing the average class distribution within the superpixel are done.

Computation of superpixels is proposed by the following method [3], in order to produce an over segmentation of the image. In this method, the pixelwise distributions d_k at superpixel k are predicted from the feature vectors Fusing a two-layer neural network [3]:

$$y_i = w_2 \tanh(w_1 F_i + b_1), \quad (3)$$

$$d_{i,a} = \frac{e^{y_{i,a}}}{\sum_{b \in \text{classes}} e^{y_{i,b}}} \quad (4)$$

$$L_{\text{cat}} = \sum_{i \in \text{pixels}} \sum_{a \in \text{classes}} d_{i,a} \ln(d_{i,a}), \quad (5)$$

$$d_{i,a} = 1/s(k) \sum d_{i,a} \quad (6)$$

where d_i is the ground truth distribution at location i, and $s(k)$ serves as the surface of the component k. Matrices W1 and W2 are trainable parameters of the classifier. Using of a two layer neural network allows the system to capture non-linear relationships between the features at different

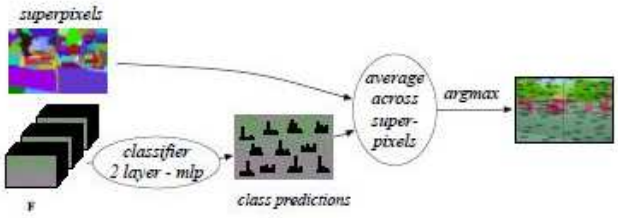

Fig. 3: Superpixel Segmentation.

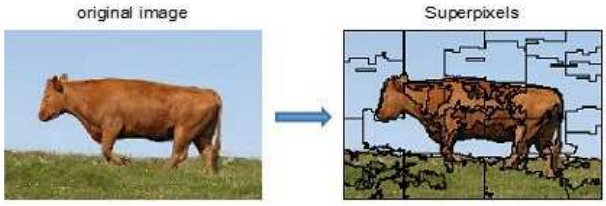

Fig. 4: Before and After Superpixel Segmentation.

scales. In this case, the final labeling for each component k is given by [3]

$$l_k = \arg\max_{a \in \text{classes}} d_{k,a} \tag{7}$$

Fig. 3 represents Eq. (7) for Superpixel Segmentation.

Fig. 4 explains the process of segmentation before and after applying Superpixel Segmentaion.

3.3 Multiscale Conditional Random Field (mCRF)

Standard CRFs employ two forms of feature functions [4], which are defined in a 2D image as follows: (i) state feature functions, $f(l_i, X, i)$, of the label at a site i and the observed image; and (ii) transition feature functions $f(l_i, l_j, X, i)$, of the image and labels at site i and a neighboring site j in the image. Label features usually encode specific patterns within the subset of label variables. The label feature is a form of potential function that encodes the specific constraints between the labels and image within the same region.

Figs. 5 and 6, describe the regional label feature with a pattern of water pixels, whereas the global label feature describes sky pixels at the top, the man pixels in the middle, and water pixels at the bottom of the image. The global features, thus can operate at a coarser resolution, specifying common value for a patch of sites in the given label field.

Along with each label feature a binary hidden variable acts as a switch for that feature. For the purpose of identifying the pattern of labels a parametrized Conditional Probability Table (CPT) is used to encode the features within the region. This CPT provides a multinomial probability distribution over the values of the

Fig. 5: Before mCRF.

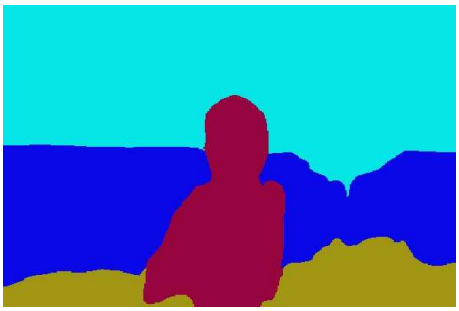

Fig. 6: Description of Region label feature.

labels in each and every site. The label variables are conditionally independent to the corresponding hidden variable and vice versa.

According to the CRF, the predictions of the features are to be combined multiplicatively. Firstly, it is not required to specify the label of every site within the same region. The combination of uniform values within the same region does not yield any significant results. Hence it is called 'don't care' prediction and it is used to predict features of particular site in the region. Secondly, the label of any site may be sharper than any of the component distributions. In this case, if two multinomial share a particular value, then that product will be significantly hiked based on the value. As a result, the unconfident predictions accomplish confident labeling.

Here mCRF [4] framework comprises three separate components, operating at three different scales: a local classifier, regional features, and global features.

3.3.1 Local Classifier

Local Classifier is used to classify the information at only the local level. The local classifier [4] produces a distribution over label variable l_i independently at each site i, provided the filter outputs x_i are within an image patch centered on pixel i [4]:

$$P_c(L \mid X, \lambda) = \pi_c(l_i \mid x_i \lambda) \tag{8}$$

where λ denotes the classifier parameter.

3.3.2 Regional Label Features

The Regional Label Features [13] denote the local geometric relationship between objects, including edges, corners or T-junctions. They specify the actual objects those are involved and avoid certain impossible combinations including a ground-above-sky border.

Let r index the regions, a index the different regional features within each region, and $j = \{1,\ldots,j\}$ index the label nodes (i.e. sites) within region r. The parameter $w_{a,j}$ connecting hidden regional variable $f_{r,a}$ and label node $l_{r,j}$ specifies the preferences for the possible label value of $l_{r,j}$. Hence $w_{a,j}$ is indicated as a vector with $|L|$ elements. The label variable $l_{r,j}$ is also represented as a vector with $|L|$ elements, in which the vth element is 1, and the other is 0 when $l_{r,j} = v$. As a result, the probabilistic model that describes regional label features comprises the following joint distribution [4]:

$$P_R(L,f) \propto \exp\left\{\sum_{r,a} f_{r,a} W_a^T i_r\right\} \tag{9}$$

where $f = \{f_{r,a}\}$ represents the binary hidden regional variables, $w_a = [w_a,1,\ldots,w_a,J,\alpha_a]$, $Ir = [lr,1,\ldots,lr,J,1]$, and α_a represents a bias term. Here the sites I are indexed by $(r;j)$, as the site i corresponds to the node j in region r which is based on the relative position of that region in the image.

3.3.3 Global Label Features

The domain of a Coarse-resolution global feature is the label field of the whole image. These global features [8] configure the undirected links between the label variables and the hidden global variables. Let b index the global label patterns encoded in the parameters $\{u_b\}$ and $g = \{g_b\}$ be the binary hidden global variables. The label field is divided into different patches in order to make these variables represent coarse aspects of the label field. These patches are non-overlapping patches $p_m m \in \{1,\ldots,M\}$, and for each hidden global variable g_b, its connections with the label nodes within patch p_m are assigned a single parameter vector $u_b p_m$. These tied parameters successfully specify the similar distribution for each label node within the patch and it also reduces the number of free parameters. Similar to the regional component, the global label feature model also has a joint distribution [8] and the Fig. 7 represents the model architecture of mCRF.

$$P_g(L,g) \propto \exp\left\{\sum_b g_b u_b^T L\right\}. \tag{10}$$

3.3.4 Combining the Components

This approach presents a model that consists of regional and global features. The present structure of this model

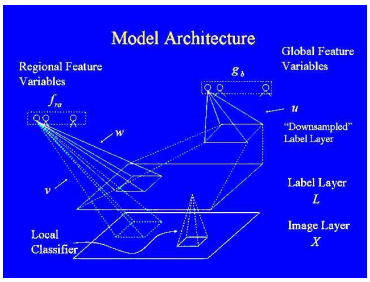

Fig. 7: Architecture of Multiscale Conditional Random Field.

permits efficient training and inference. Apart from that, these label features form a redundant requirements of label predictions that can be combined multiplicatively. Based on these reasons, the features encode simple, geometric relationships that exist between specific label classes. The multiplicatively combined probability distribution over the label field has a simple closed form [4]

$$P(L \mid x;\theta) = 1/z\pi_c p_c^{\gamma}(l_i \mid x_i,\lambda)x \tag{11}$$

$$X\pi_{r,a}[1+\exp(w_a^T l_r)]X\pi_b[1+\exp(u_b^T L)] \tag{12}$$

where $\theta = \{\lambda,\{w_a\},\{u_b\},\gamma\}$ is the set of parameters in the model.

3.3.5 Features of MCRF

The regional and global label features used by the local classifier do not have any access to image statistics. This results in the assumption that in this model the context is independent of any local evidence. It will be interesting to train features that have access to local image statistics. The label features, in this model, are defined at two different scales. There is a continuity from local to global context in mCRF in reality. One solution is to spread out features over different scales and this may be doubtful to be the optimal solution. If label features are automatically learnt over the optimal scales from the labeled data, this may yield a better representation of context in the given image. If two different scales are fixed it might enhance the model's ability to classify objects which occurs consistently in a small scale, for example, a mouse or a keyboard. Through this approach the relationship between the object class and the features is revealed. With the increase of the object classes the process of capturing the feature also increase proportionately. Hence the features presents the geometric relationship among the object classes.

3.4 Experimental Result

The proposed method is tested on two different fully labeled datasets: the Stanford Background and the SIFT

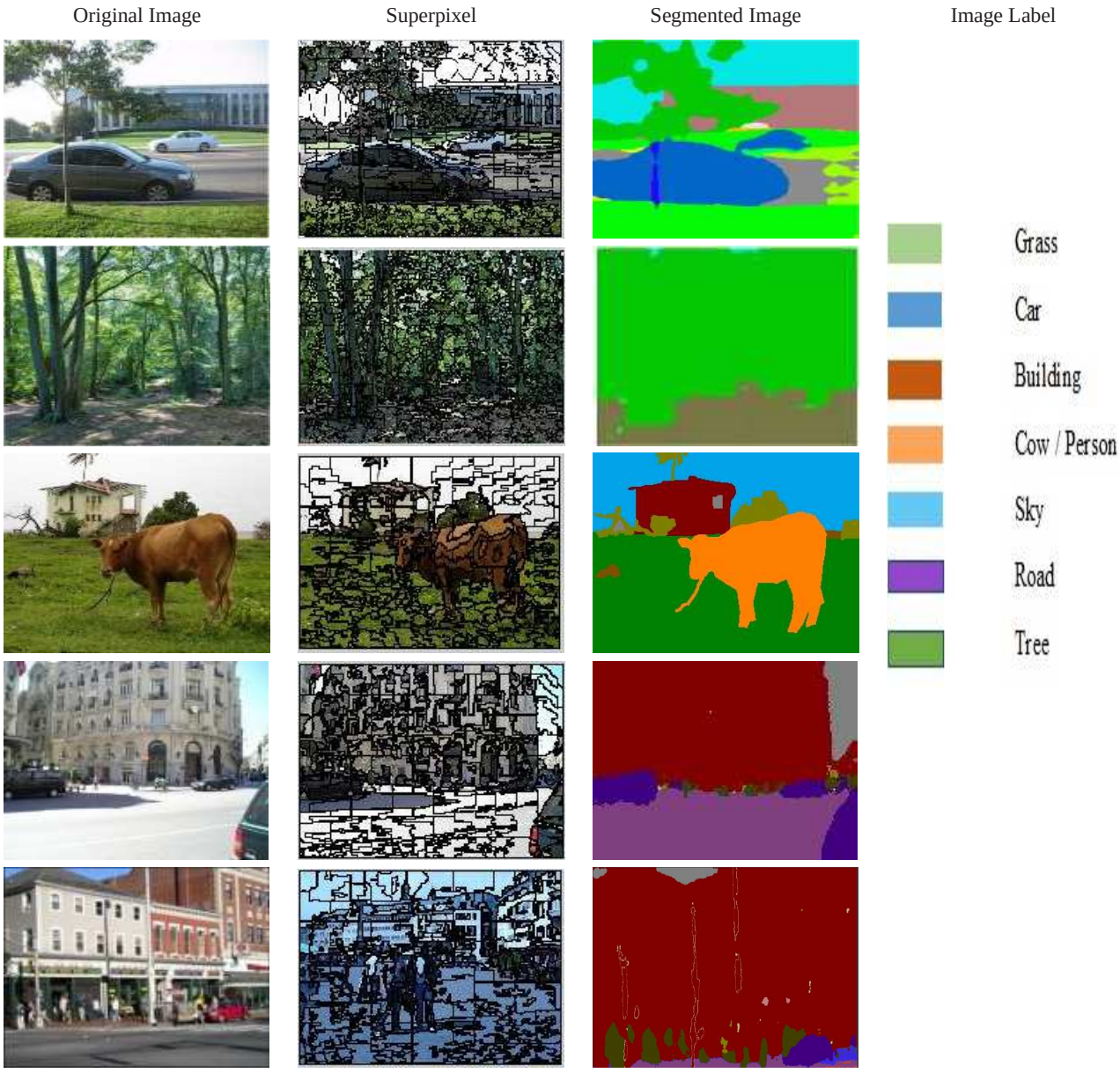

Fig. 8: Experimental output of the proposed approach.

Flow Dataset [13]. The Stanford dataset has 715 images from rural and urban scenes comprising 8 classes. The scenes have approximately 320×240 pixels. The SIFT Flow is a larger dataset which consists of 2688 images with 256×256 pixels and 33 semantic labels.

All these networks were trained by sampling patches which are surrounded by a pixel which is chosen randomly from a randomly chosen image from the training image set. There are two different approaches followed to find out the accuracy of the image. They are (i) Pixel-wise accuracy (ii) Class-wise accuracy. Pixel-wise accuracy indicates the ratio of pixels which are correctly predicted, while class-wise IoU indicates the Intersection of Union of pixels averaged over all the 150 semantic categories. However, in scene labeling (especially in datasets with large number of classes), classes which are much more frequent than others (e.g. the class 'sky' is much more frequent than 'moon') have more impact on this measure.

Some random images from Stanford and Sift Flow dataset are tested with this approach as in Fig. 8. First, the feature of the whole image is extracted exploiting the ReNet, which in turn, is pre-segmented using Superpixels and multiscale Conditional Random Field. The purpose of pre-segmentation is to identify and explore whether some pixels belong to one group or not. In the pre-segmentation

Table 1: Accuracy Comparison.

Methods	Explanation	Pixel Accuracy (%)	Comp Time (S)
Region based Model	17-Dimensional Color and Texture Features, 9 Grid locations around thePixel and the image row,Region Segmentation [14]	76.4	10–600
SuperParsing	Global, Shape, Location, Texture/Sift, Color, Appearance, MRF [15]	77.5	10–300
Stacked Hierarchical Labeling	Gist, Pyramid Histogram of Oriented Gradients, Color Histogram Cielab, Relative relocation, Hierarchical region representation.	76.9	12
Relationship Prediction Model	Color, Texture, Shape, Percentage pixels above horizontal, Region-based Segmentation [16]	79.4	< 600
Learning Hierarchical Features	Laplacian Pyramid, Superpixels/CRF/ Tree Segmentation, Data augmentation	78.8	0.6
Our Approach	ReNet Architecture + Superpixels + Multiscale CRF	80.2	11

layer, the average score of a pixel is computed, which in turn is assigned to each and every pixel belonging to the same group. In Fig. 8, the experimental output of the proposed approach for the random images for both data set are given.

Table 1 explicate the results based on the comparison between the proposed and other existing approaches with reference to pixel accuracy and computation time. When comparing all the methods with our approach the pixel accuracy is high but the computation time is little bit higher. It is because of the preprocessing methods which is applied before labeling.

4 Conclusion

This paper uses the ReNet architecture and pre-segmentation methods which include Superpixels and Multiscale Conditional Random Field to improve the Scene labeling strategy. The results of the experiment prove that the proposed method provides higher pixel-accuracy when compared with other methods. Using the simpler architecture without any pre-segmentation techniques leads to lesser computing time per image. This proposed approach uses pre-segmentaion methods, which in turn increases the computing time per image, but at the same time, it improves the accuracy when compared to other existing methods. This framework has proved to be a successful method to detect the objects in natural scenes, more effectively in analyzing the images and comparing their presence. As a future scope, the method can be modified without using these additional steps at the same time reducing the running time. Moreover this paper deals with some of the random images of the Sift Flow and Stanford dataset only. In future it can be extended to full dataset and for the datasets which contain more images like Barcelona Dataset.

References

[1] Qingsong Zhu, Guanzheng Liu, Zhanyong Mei, Yaoqin Xie and Lei Wang, Perfect Snapping: An Accurate and Efficient Interactive Image Segmentation Algorithm, International Journal of Applied Mathematics and Information Sciences, 1387–1393, 2013.

[2] Francesco Visin, Kyle Kastner, Kyunghyun Cho, Matteo Matteucci, Aaron Courville, Yoshua Bengio, ReNet: A Recurrent Neural Network Based Alternative to Convolutional Networks, Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2015.

[3] Junjie Yan, Yinan Yu, Xiangyu Zhu, Zhen Lei and Stan Z. Li, Object detection by Labeling Superpixels, In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Oct 2015.

[4] Xuming He, Richard S. Zemel, and Miguel A. Carreira-Perpinãn, Multiscale Conditional Random Fields for Image Labeling, in CVPR'04 Proceedings of the 2004 IEEE computer society conference on Computer vision and pattern recognition, 695–703.

[5] Pedro O. Pinheiro and Ronan Collobert, Recurrent Convolutional Neural Networks for Scene Labeling, International Conference on Machine Learning, Beijing, China, 2014. JMLR: W&CP volume 32.

[6] Jiang Wang, Yi Yang, Junhua Mao, Zhiheng Huang, Chang Huang, Wei Xu1, CNN- RNN: A Unified Framework for Multi-label Image Classification, in CVPR Proceedings of the 2016 IEEE computer society conference on Computer vision and pattern recognition, 2016.

[7] C. Farabet, C. Couprie, L. Najman, Y. LeCun, Scene Parsing with Multiscale Feature Learning, Purity Trees, and Optimal Covers, In: Proc. of the International Conference on Machine Learning (ICML'12), June 2012.

[8] V. Lempitsky, A. Vedaldi, and A. Zisserman, A pylon model for semantic segmentation, In Advances in Neural Information Processing Systems, 2011.

[9] R. Socher, C. C. Lin, A. Y. Ng, and C. D. Manning, Parsing Natural Scenes and Natural Language with Recursive Neural Networks, In Proceedings of the 26th International conference on Machine Learning (ICML), 2011.

[10] D. Grangier, L. Bottou, and R. Collobert, Deep Convolutional Networks for Scene Parsing, In ICML 2009 Deep Learning Workshop, 2009.
[11] C. Russell, P.H.S. Torr, and P. Kohli, Associative hierarchical CRFs for object class image segmentation, In Proc. ICCV, 2009.
[12] D. Munoz, J. Bagnell, and M. Hebert, Stacked hierarchical labeling, ECCV 2010, Jan 2010.
[13] Ce Liu, Jenny Yuen and Antonio Torralba, SIFT Flow: Dense Correspondence across Scenes and its Applications, IEEE Transactions on Pattern Analysis and Machine Intelligence, 2011.
[14] S. Gould, R. Fulton, and D. Koller, Decomposing a scene into geometric and semantically consistent regions, IEEE International Conference on Computer Vision, pages 1–8, Sept. 2009.
[15] J. Tighe and S. Lazebnik, Superparsing: scalable nonparametric image parsing with superpixels, ECCV, 352–365, 2010.
[16] M. Kumar and D. Koller, Efficiently selecting regions for scene understanding, In Computer Vision and Pattern Recognition (CVPR), 3217–3224. IEEE, 2010.

N. Shanmugapriya is working as an Associate Professor in department of IT, Oxford Engineering college, Trichy. She is pursuing her doctorate in Anna University, Chennai and obtained her M.E. in Computer Science and Engineering. She has published 6 papers so far in International and national Conferences and Journals. She has Guided 15 projects in both under graduate and postgraduate students towards their project work. She has 12 years of teaching experience. Her area of interest is Image Processing and Pattern Recognition.

D. Chitra is working as a Professor and Head in the department of CSE, P.A. College of Engineering and Technology. She received her Doctor of Philosophy from Anna University, Chennai and Master's Degree in Computer Science and Engineering. Her areas of interest include Digital Image Processing, Pattern Recognition, Computer Vision, Data Mining and Grid & Cloud Computing. She has 17 years of experience in teaching and published 70 papers in National and International Conferences and Journals. She is a member of IEEE, ISTE, CSI, IAENG and IRED. She has guided 67 projects in both UG and PG, and currently 9 research scholars pursuing Ph.D. She is a reviewer for many Journals and Conferences. She attended 25 national and International seminars/conferences/workshops. She has received awards such as Best Circuit Faculty Award SIAA (ASDF), Shri. P.K. Das Best Faculty Award, Best Faculty Award in Kongu Engineering College and Best Faculty Award in P.A. College of Engineering and Technology. She also organized 21 programmes sponsored by AICTE, Anna University, CSIR, DRDO, ICMR, and INSA.