Scene Comprehension Through Image Analysis with an Extensive Array of Categories and Context at the Scene Level

Abstract

This research introduces a unique approach to scene parsing that is nonparametric, which enhances the precision and expands the scope of foreground categories within images of scenes. Initially, the accuracy of label likelihood at the superpixel level is improved by combining likelihood scores from multiple probabilistic classifiers. This method improves classification accuracy and enhances the representation of categories that are less frequently represented. The second advancement involves the integration of semantic context into the parsing procedure by utilizing global label costs. Instead of relying on sets derived from image retrieval, the technique described assigns a comprehensive likelihood estimate to each label, which is subsequently incorporated into the overall energy function. The effectiveness of the system is assessed using two expansive datasets, SIFTflow and LMSun. The system demonstrates performance that is at the forefront of the field on the SIFTflow dataset and achieves outcomes that are close to setting new records on the LMSun dataset.

1 Introduction

The task of scene parsing involves assigning semantic labels to every pixel within an image of a scene. Algorithms for image parsing attempt to categorize different types of scenes, both indoors and outdoors, such as a shoreline, a roadway, an urban environment, and an airport. Numerous systems have been developed to categorize each pixel in an image semantically. A significant obstacle for image parsing methods is the considerable variability in recognition rates across different types of classes. Background classes, which usually cover a significant area of the image's pixels, often have a uniform look and are identified with great accuracy. Foreground classes, which usually take up fewer pixels in the image, have changeable forms and might be hidden or set up in various ways. These kinds of classes represent noticeable parts of the image that frequently grab a viewer's attention. However, their recognition rates are often much lower than those of background classes, making them frequent examples of unsuccessful recognition.

Impressive results have been obtained by parametric scene parsing techniques on datasets with a limited number of labels. Nevertheless, for considerably bigger datasets with a lot of labels, using these techniques becomes more challenging because of the increased demands on learning and optimization.

Nonparametric image parsing techniques have recently been introduced to tackle the growing variety of scene types and semantic labels effectively. These methods usually begin by reducing the complexity of the problem from individual pixels to superpixels. Initially, a set of images is selected, consisting of training images that bear the closest visual resemblance to the image being queried. The potential labels for a specific image are limited to those found in the selected set of images. Subsequently, the probability scores for the classification of superpixels are determined by matching visual characteristics. Ultimately, context is applied by reducing an energy function that includes both the expense of the data and information on how often classes appear together in nearby superpixels.

.

A shared difficulty encountered by nonparametric parsing methods is the phase of image retrieval. Even though image retrieval helps narrow down the number of labels to think about, it's seen as a very important step in the process. There's no opportunity to correct the mistake if the correct labels are not among the images that were retrieved. It has been reported that mistakes in retrieval are the main reason for most unsuccessful cases.

A novel nonparametric image parsing algorithm is proposed in this work, aiming for enhanced overall precision and improved identification rates for classes that are less commonly represented. An efficient system is developed that can adapt to an ever-growing quantity of labels. The contributions made are outlined as follows:

1. Superpixel label likelihood scores are improved by merging classifiers. The system merges the output probabilities from several classification models to generate a more equitable score for each label at every superpixel. The weights for merging the scores are determined by employing a likelihood normalization technique on the training set in an automated manner. 2. Semantic context is integrated within a probabilistic structure. To prevent the removal of important labels that cannot be retrieved later, a retrieval set is not structured. Instead, label costs are utilized, which are determined from the global contextual relationships of labels in analogous scenes, to obtain enhanced parsing outcomes.

The system developed achieves top-tier per-pixel recognition accuracy on two extensive datasets: SIFTflow, which includes 2688 images with 33 labels, and LMSun, which has 45576 images with 232 labels.

2 Related Work

Several techniques for scene parsing, both parametric and nonparametric, have been suggested. The nonparametric systems that try to cover a wide range of semantic classes are very similar to the method. Different methods are used to improve the overall effectiveness of nonparametric parsing. The authors merge region-parsing with outputs from per-exemplar SVM detectors. Object masks are transferred by per-exemplar detectors into the test image for segmentation. Their method greatly improves overall accuracy, but it requires a lot of computer power. It's hard to scale because data terms need to be calibrated using a batch of fine training in a leave-one-out way, which is hard to do. Superpixels from rare classes are specifically added to the retrieval set to make them more visible. The authors filter the list of labels for a test image by doing an image retrieval step, and query time is used to add more samples to rare classes. The way superpixels are classified, how rare classes are recognized, and how semantic context is applied are all different in this system. By combining classification costs from different contextual models, a more balanced set of label costs is produced, which promotes the representation of foreground classes. Instead of using image retrieval, global label costs are used in the inference step.

The value of semantic context has been thoroughly investigated in numerous visual recognition algorithms. Context has been employed to enhance the overall labeling performance through a feedback mechanism in nonparametric scene parsing systems. Initial labeling of superpixels in a query image is utilized to modify the training set by adjusting for recognized background classes, thereby enhancing the visibility of uncommon classes. The objective is to enhance the image retrieval set by reintroducing segments of uncommon classes. A semantic global descriptor is generated. Image retrieval is enhanced by merging the semantic descriptor with the visual descriptors. Context is added by creating global and local context descriptors based on classification likelihood maps. The method described differs from these methods as it does not employ context at each superpixel when calculating a global context descriptor. Instead, contextual information across the entire image is taken into account.

Contextually relevant outcomes are produced by deducing label correlations in comparable scene images. Additionally, there is no retrieval set that needs to be enriched. Rather, the global context is structured within a probabilistic framework, where label costs are calculated across the whole image. Furthermore, the global context is executed in real time without any preliminary training. Another method of image parsing that doesn't use retrieval sets is where image labeling is done by moving annotations from a graph of patch matches across image sets. But this method needs a lot of memory, which makes it hard to scale for big datasets.

The presented method draws inspiration from the combination of classifier techniques in machine learning, which have demonstrated the ability to enhance the capabilities of individual classifiers. Several fusion methods have been effectively applied in various fields of computer vision, including detecting faces, annotating images with multiple labels, tracking objects, and recognizing characters. Nonetheless, the classifiers that make up these systems and the ways they are combined are very different from the framework, and the other methods have only been tested on small datasets.

3 Baseline Parsing Pipeline

This section provides a summary of the basic image parsing system, which is composed of three stages: feature extraction, label likelihood estimation at superpixels, and inference.

Afterward, contributions are presented: enhancing likelihoods at superpixels and calculating label costs for global context at the scene level.

3.1 Segmentation and Feature Extraction

To reduce the complexity of the task, the image is partitioned into superpixels. Extraction of superpixels from images begins by employing an efficient graph-based method. For each superpixel, 20 distinct types of local features are extracted to characterize its shape, appearance, texture, color, and position, adhering to established methods. In addition to these features, Fisher Vector (FV) descriptors are extracted at each superpixel using an established library. Computation of 128-dimensional dense SIFT feature descriptors is performed on five different patch sizes (8, 12, 16, 24, 30). A dictionary comprising 1024 words is constructed. Subsequently, the FV descriptors are retrieved and Principal Component Analysis (PCA) is applied to decrease their dimensionality to 512. Each superpixel is represented by a feature vector that has 2202 dimensions.

3.2 Label Likelihood Estimation

The features obtained in the prior stage are utilized to determine label probabilities for each superpixel. Unlike conventional approaches, the possible labels for a test image are not restricted. Instead, the data term for the likelihood of each class label c C is computed, where C represents the total number of classes in the dataset. The normalized cost D(l_{si} = c|s_i) of assigning label c to superpixel s_i is given by:

$$D(l_{si} = c|s_i) = 1 - \frac{1}{1 + e^{-L_{unbal}(s_i,c)}} \tag{1}$$

where L_{unbal}(s_i, c) is the log-likelihood ratio score of label c, given by L_{unbal}(s_i, c) = 1/2 log(P(s_i|c)/P(s_i|¬c)), where ¬c = C c is the set of all labels except c, and P(s_i|c) is the likelihood of superpixel s_i given c. A boosted decision tree (BDT) model is trained to obtain the label likelihoods L_{unbal}(s_i, c). For implementation, a publicly accessible boostDT library is utilized. During this phase, the BDT model is trained using every superpixel in the training set, which constitutes an imbalanced distribution of class labels C.

3.3 Smoothing and Inference

The optimization challenge is formulated as a maximum a posteriori (MAP) estimation to determine the ultimate labeling L through Markov Random Field (MRF) inference. Using only the estimated likelihoods from the preceding section to categorize superpixels leads to imprecise classifications. Incorporating a smoothing term V(l_{s_i}, l_{s_j}) into the MRF energy function aims to address this problem by penalizing adjacent superpixels with semantically incongruous labels. The goal is to minimize the following energy function:

$$E(L) = \sum_{s_i \in S} D(l_{s_i} = c|s_i) + \lambda \sum_{(i,j) \in A} V(l_{s_i}, l_{s_j}) \tag{2}$$

where A represents the set of neighboring superpixel indices and V(l_{s_i}, l_{s_j}) denotes the penalty for assigning labels l_{s_i} and l_{s_j} to two adjacent pixels, calculated from occurrences in the training set combined with the constant Potts model following established methods. is the smoothing constant. Inference is conducted using the -expansion method with established code.

4 Improving Superpixel Label Costs

Although foreground objects typically stand out the most in a picture of a scene, parsing algorithms frequently misclassify them. For instance, in an image of a city street, a person would usually first spot the individuals, signs, and vehicles before they would see the structures and the street. However, because of two primary factors, scene parsing algorithms frequently misclassify foreground regions as belonging to the surrounding background. Initially, in the superpixel classification phase, any classifier would naturally prefer classes that are more prevalent to reduce the overall training error. Secondly, during the MRF smoothing phase, a lot of the superpixels that were accurately identified as foreground objects are smoothed out by the background pixels around them.

It is suggested that the label likelihood score at each superpixel be improved to obtain a more precise parsing output. Various classifiers are designed that provide supplementary information regarding the data. Subsequently, all the developed models are merged to produce a unified conclusion. An overview of the method for merging classifiers is displayed in Figure 1. During the testing phase, the label likelihood scores from all the BDT models are combined to generate the final scores for superpixels.

4.1 Fusing Classifiers

The proposed method is inspired by ensemble classifier methods, which train several classifiers and merge them to enhance decision-making. These methods are especially helpful when the classifiers are distinct. In other words, the decrease in error is connected to the lack of correlation between the models that were trained. This means that the total error is decreased if the classifiers misclassify different data points. Furthermore, it has been demonstrated that for large datasets, dividing the training set yields superior results compared to dividing the feature space.

It has been observed that the classification error for a particular class is correlated with the average number of pixels it covers in the scene images, as indicated by the blue line in Figure 2. This is in line with what earlier methods found, which is that the rate of classification error is related to how often classes show up in the training set. However, it goes beyond that by taking into account how often the classes appear at the image level, which is meant to solve the problem of less-represented classes being smoothed out by a background class that is nearby.

To achieve this, three BDT models are trained using the following training data criteria: (1) a balanced subsample of all classes C in the dataset, (2) a balanced subsample of classes that occupy an average of less than z

The goal of these decisions is to lessen the correlation between the trained BDT models, as seen in Figure 2. The balanced classifiers are able to correctly identify some of the less-represented classes, but they make more mistakes on the more-represented classes. The unbalanced classifier, on the other hand, mostly misclassifies the less-represented classes. Combining the likelihoods from all the classifiers leads to an improved overall decision that enhances the representation of all classes (Figure 1). It was noticed that the addition of more classifiers did not enhance performance for any of the datasets.

The ultimate expense of allocating a label c to a superpixel s_i can subsequently be expressed as the amalgamation of the likelihood scores of all classifiers:

$$D(l_{si} = c|s_i) = 1 - \frac{1}{1 + e^{-L_{comb}(s_i,c)}} \tag{3}$$

where L_{comb}(s_i, c) represents the combined likelihood score obtained by the weighted sum of the scores from all classifiers:

$$L_{comb}(s_i, c) = \sum_{j=1,2,3,4} w_j(c) L_j(s_i, c) \tag{4}$$

where L_j(s_i, c) is the score from the jth classifier, and w_j(c) is the normalized weight of the likelihood score of class c in the jth classifier.

4.2 Normalized Weight Learning

The weights w

$w < sub > j < /sub > (c)]arelearnedforallclassesCinoflinesettingsusingthetrainingset.Theweightsarecalculate$

$$\tilde{w}_j(c) = \frac{1}{|C_j|} \frac{\sum_{s_i \in S} L_j(s_i, c)}{\sum_{c_i \in C \setminus c} \sum_{s_i \in S} L_j(s_i, c_i)} \tag{5}$$

where |C_j| denotes the quantity of classes encompassed by the jth classifier and not covered by any other classifier with a fewer number of classes.

The normalized weight w_j(c) of class c can then be computed as: w_j(c) = ~w_j(c) / _{j=1,2,3,4}(~w_j(c)). Normalizing the output likelihoods in this way improves the likelihood that all classifiers will be taken into account in the outcome, with a focus on classes that are less represented.

5 Scene-Level Global Context

When working with scene parsing challenges, including the scene's semantics in the labeling process is beneficial. For example, if a scene is known to be a beach scene, labels such as sea, sand, and sky are expected to be found with a much greater probability than labels like car, building, or fence. The initial labeling results of a test image are used in estimating the likelihoods of all labels c C. The likelihoods are estimated globally over an image, i.e., there is a unique cost per label per image. The global label costs are then incorporated into a subsequent MRF inference stage to enhance the results.

The presented method, in contrast to previous methods, does not restrict the number of labels to those found in the retrieval set. Instead, it utilizes the set to calculate the likelihood of class labels in a k-nn manner. The likelihoods are normalized by counts over the entire dataset and smoothed to provide an opportunity for labels not present in the retrieval set. The likelihoods are also used in MRF optimization, not for reducing the number of labels.

5.1 Context-Aware Global Label Costs

It is proposed that semantic context be incorporated by using label statistics instead of global visual features. The reasoning behind this decision is that sorting by global visual characteristics often doesn't find images that are similar at the scene level. For instance, a highway scene might be mistaken for a beach scene if road pixels are incorrectly classified as sand. Nonetheless, when given a reasonably accurate initial labeling, sorting by label statistics finds images that are more semantically related. This helps to eliminate outlier labels and find labels that are absent in a scene.

For a given test image I, minimizing the energy function in equation 2 produces an initial labeling L of the superpixels in the image. If C is the total number of classes in the dataset, let T C be the set of unique labels which appear in L, i.e. T = t | s_i : l_{s_i} = t, where s_i is a superpixel with index i in the test image, and l_{s_i} is the label of s_i. Semantic context is exploited in a probabilistic framework, where the conditional distribution P(c|T) is modeled over class labeling C given the initial global labeling of an image T. P(c|T) c C is computed in a K-nn fashion:

$$P(c|T) = \frac{1 + n(c, K_T)}{n(c, S)} \frac{1 + n(\neg c, K_T)}{|S|} \tag{6}$$

where K_T is the K-neighborhood of initial labeling T, n(c, X) is the number of superpixels with label c in X, n(¬c, X) is the number of superpixels with all labels except c in X, and |S| is the total number of superpixels in the training set. The likelihoods are normalized and a smoothing constant of value 1 is added.

To obtain the neighborhood K_T, training images are ranked by their distance to the query image. The distance between two images is determined by the weighted size of the intersection of their class labels, which intuitively shows that the neighbors of T are images that share many labels with those in T. A different weight is assigned to each class in T in a manner that gives preference to classes that are less represented.

The algorithm operates in three stages, as depicted in Figure 3. It begins by (1) assigning a weight _t to each class t T, which is inversely proportional to the number of superpixels in the test image with label t: _t = 1 - n(t,I)/|I|, where n(t, I) is the number of superpixels in the test image with label l_{s_i} = t, and |I| is the total number of superpixels in the image. Then, (2) training images are ranked by the weighted size of intersection of their class labels with the test image. Finally, (3) the global label likelihood L_{global}(c) = P(c|T) of each label c C is computed using equation 6.

Calculating the label costs is performed in real-time for a query image, without the need for any offline batch training. The method enhances the overall precision by utilizing solely the true labels of training images, without incorporating any global visual characteristics.

5.2 Inference with Label Costs

Once the likelihoods L_{global}(c) of each class c C are obtained, a label cost H(c) = -log(L_{global}(c)) can be defined. The final energy function becomes:

$$E(L) = \sum_{s_i \in S} D(l_{s_i} = c | s_i) + \lambda \sum_{(i,j) \in A} V(l_{s_i}, l_{s_j}) + \sum_{c \in C} H(c)\delta(c) \tag{7}$$

where (c) is the indicator function of label c:

$$\delta(c) = \begin{cases} 1 & \text{if } \exists s_i : l_{s_i} = c \\ 0 & \text{otherwise} \end{cases} \tag{8}$$

Equation 7 is solved using -expansion with the extension method to optimize label costs. Optimizing the energy function in equation 7 effectively minimizes the number of unique labels in a test image to those with low label costs, i.e., those most relevant to the scene.

6 Experiments

The experiments were conducted on two extensive datasets: SIFTflow and LMSun. SIFTflow consists of 2,488 training images and 200 test images. All images are of outdoor scenes, sized 256x256 with 33 labels. LMSun includes both indoor and outdoor scenes, with a total of 45,676 training images and 500 test images. Image sizes range from 256x256 to 800x600 pixels with 232 labels.

The same evaluation metrics and train/test splits as in previous methods are employed. The per-pixel accuracy (the percentage of pixels in test images that were correctly labeled) and per-class recognition rate (the average of per-pixel accuracies of all classes) are reported. The following variants of the system are evaluated: (i) baseline, as described in section 3, (ii) baseline (with balanced BDT), which is the baseline approach using a balanced classifier, (iii) baseline + FC (NL fusion), which is the baseline in addition to the fusing classifiers with normalized-likelihood (NL) weights in section 4, and (iv) full, which is baseline + fusing classifiers + global costs. To show the effectiveness of the fusion method (section 4.2), the results of (v) baseline + FC (average fusion), which is fusing classifiers by averaging their likelihoods, and (vi) baseline + FC (median fusion), which is fusing classifiers by taking the median of their likelihoods are reported. Results of (vii) full (without FV), which is the full system without using the Fisher Vector features are also reported.

x = 5 is fixed (section 4.1), a value that was obtained through empirical evaluation on a small subset of the training set.

6.1 Results

The results are compared with state-of-the-art methods on SIFTflow in Table 1. K = 64 top-ranked training images have been set for computing the global context likelihoods (section 5.1). The full system achieves 81.7

Table 1: Comparison with state-of-the-art per-pixel and per-class accuracies (%) on the SIFTflow dataset.

Method	Per-pixel	Per-class
Liu et al.	76.7	N/A
Farabet et al.	78.5	29.5
Farabet et al. balanced	74.2	46.0
Eigen and Fergus	77.1	32.5
Singh and Kosecka	79.2	33.8
Tighe and Lazebnick	77.0	30.1
Tighe and Lazebnick	78.6	39.2
Yang et al.	79.8	48.7
Baseline	78.3	33.2
Baseline (with balanced BDT)	76.2	45.5
Baseline + FC (NL fusion)	80.5	48.2
Baseline + FC (average fusion)	78.6	46.3
Baseline + FC (median fusion)	77.3	46.8
Full without Fisher Vectors	77.5	47.0
Full	81.7	50.1

Table 2 compares the performance of the same variants of the system with the state-of-the-art methods on the large-scale LMSun dataset. LMSun is more challenging than SIFTflow in terms of the number of images, the number of classes, and the presence of both indoor and outdoor scenes. Accordingly, a larger value of K = 200 in equation 6 is used. The method achieves near-record performance in per-pixel accuracy (61.2

Table 2: Comparison with state-of-the-art per-pixel and per-class accuracies (%) on the LMSun dataset.

Method	Per-pixel	Per-class
Tighe and Lazebnick	54.9	7.1
Tighe and Lazebnick	61.4	15.2
Yang et al.	60.6	18.0
Baseline	57.3	9.5
Baseline (with balanced BDT)	45.4	13.8
Baseline + FC (NL fusion)	60.0	14.2
Baseline + FC (average fusion)	60.5	11.4
Baseline + FC (median fusion)	59.2	14.7
Full without Fisher Vectors	58.2	13.6
Full	61.2	16.0

The performance of the system is analyzed when varying the number of trees T for training the BDT model (section 4.1), and the number of top training images K in the global label costs (section 5.1). Figure 4 shows the per-pixel accuracy (on the y-axis) and the per-class accuracy (on the x-axis) as a function of T for a variety of K's. Increasing the value of T generally produces better classification models that better describe the training data. At T 400, performance levels off. As shown, the global label costs consistently improve the performance over the baseline method with no global context. Using more training images (higher K) improves the performance through considering more semantically relevant scene images. However, performance starts to decrease for very high values of K (e.g., K = 1000) as more noisy images start to be added.

Figure 5 shows the per-class recognition rate for the baseline, combined classifiers, and the full system on SIFTflow. The fusing classifiers technique produces more balanced likelihood scores that cover a wider range of classes. The semantic context step removes outlier labels and recovers missing labels, which improves the recognition rates of both common and rare classes. Recovered classes include field, grass, bridge, and sign. Failure cases include extremely rare classes, e.g. cow, bird, desert, and moon.

6.2 Running Time

The runtime performance was analyzed for both SIFTflow and LMSun (without feature extraction) on a four-core 2.84GHz CPU with 32GB of RAM without code optimization. For the SIFTflow dataset, training the classifier takes an average of 15 minutes per class. The training process is run in parallel. The training time highly depends on the feature dimensionality. At test time, superpixel classification is efficient, with an average of 1 second per image. Computing global label costs takes 3 seconds. Finally, MRF inference takes less than one second. MRF inference is run twice for the full pipeline. LMSun is much larger than SIFTflow. It takes 3 hours for training the classifier, less than a minute for superpixel classification per image, less than 1 minute for MRF inference, and 2 minutes for global label cost computation.

6.3 Discussion

The presented scene parsing method is generally scalable as it does not require any offline training in a batch fashion. However, the time required for training a BDT classifier increases linearly with increasing the number of data points. This is challenging with large datasets like LMSun. Randomly subsampling the dataset has a negative impact on the overall precision of the classification results. Alternative approaches of mining discriminative data points that better describe each class are planned to be investigated. The system still faces challenges in trying to recognize very less-represented classes in the dataset (e.g., bird, cow, and moon). This could be handled via better contextual models per query image.

7 Conclusion

A novel scene parsing algorithm has been presented that enhances the overall labeling precision, without neglecting foreground classes that are significant to human viewers. By merging likelihood scores from various classification models, the strengths of individual models have been successfully amplified, thus enhancing both the per-pixel and per-class accuracy. To prevent the removal of accurate labels through image retrieval, global context has been integrated into the parsing process using a probabilistic framework. The energy function has been expanded to incorporate global label costs that produce a more semantically relevant parsing output. Experiments have demonstrated the superior performance of the system on the SIFTflow dataset and comparable performance to state-of-the-art methods on the LMSun dataset.