Unsupervised Patch-based Context from Millions of Images

Santosh Divvala[1], Alexei Efros[1], Martial Hebert[1], and Svetlana Lazebnik[2]

[1] Robotics Institute, Carnegie Mellon University.
[2] University of North Carolina at Chappel Hill.

Abstract. The amount of labeled training data required for image interpretation tasks is a major drawback of current methods. How can we use the gigantic collection of unlabeled images available on the web to aid these tasks? In this paper, we present a simple approach based on the notion of *patch-based context* to extract useful priors for regions within a query image from a large collection of (6 million) unlabeled images. This contextual prior over image classes acts as a non-redundant complimentary source of knowledge that helps in disambiguating the confusions within the predictions of local region-level features. We demonstrate our approach on the challenging tasks of region classification and surface-layout estimation.

1 Introduction

Image interpretation deals with the problem of parsing an image of a scene into its constituent regions. This problem is often posed as a multi-class region classification task, i.e., associating a label to every pixel within an image. Many approaches have been proposed recently to address this problem e.g., [1–5]. Almost all of the approaches confine to the use of a small set of labeled images for modeling the region classes. However, for most practical problems (with many classes and high variability within the classes) there is simply not enough labeled data available to learn rich discriminative models for classification. Although with the rise of the Amazon Mechanical Turk and other online collaborative annotation efforts, the process of gathering more labeled data has been greatly eased for several tasks, pixel-wise image labeling remains difficult as it is much more involved. Thanks to the popularity of social-networking and photo-sharing websites (Facebook, Flickr), today there exist several billions of unlabeled images available on the web. Devising algorithms that can automatically benefit from this wealth of information can help alleviate the labeled data barrier.

Since the images on the web are unlabeled, their effective use in learning better models for classification is not straightforward. A lot of research exists in the area of semi-supervised machine learning to specifically deal with this problem e.g., transductive support vector machines, graph-based semi-supervised learning, co-training (see [6] for literature survey). The key idea of these methods is to exploit labeled samples as well as a large number of unlabeled samples for obtaining an accurate decision boundary. This intuition is summarized by the so-called "cluster assumption", i.e., provided different classes come in clearly separated clusters, unlabeled data can help to delineate the boundaries of the

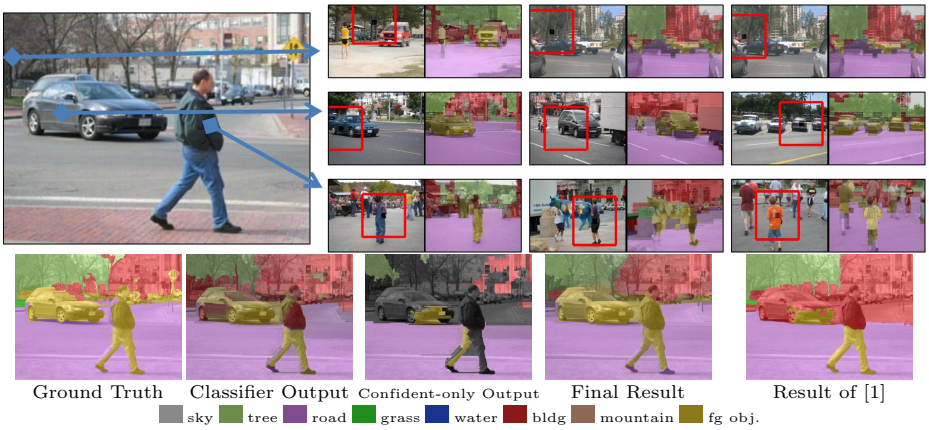

Fig. 1. Given a query image, we retrieve matches for each individual (local) patch by searching a database of 6 million images. The matches are used to compute contextual priors that are used in updating a supervised classifier (trained from a small set of labeled images) to improve its performance.

clusters better [7]. For the problem of image parsing though, the "cluster assumption" fails to hold due to high appearance ambiguity, i.e., regions that look very similar in terms of appearance can have different labels. This ambiguity problem has prevented the conventional semi-supervised methods from succeeding at this task [8]. One possible solution to overcome the ambiguity in features is to increase the size of the elementary patch units used in the clustering process. However this is not plausible due to the scarcity of labeled data. That is, we would end up with well separated clusters but with no labeled data samples to label them.

To address this trade-off, we propose to decouple the patch size and the data types involved. As there is only limited amount of labeled data, we use a small patch size, which provides us with enough data to learn a rich local classifier. While for the unlabeled data, which is available in large quantities, we use a larger patch which would allow the 'cluster assumption' to hold and would encode longer range connections not accessible to the local classifier. It must be noted that the idea of using a larger neighborhood to guide the local classifier has been studied in many recent works [1, 2, 4, 9–14, 5]. The basic insight shared amongst them is to use information from the neighborhood around a local patch to derive prior probability over different classes for it. However, almost all of the approaches have considered deriving such priors in completely supervised settings i.e., using label information from annotated images, which as highlighted earlier comes in scarce quantity and thus prevents from learning anything at a larger scale.

In this paper, we present an approach for deriving unsupervised patch-based context from a large collection (millions) of unlabeled Internet images for the tasks of region classification [1] and surface-layout estimation [2]. In section 2, we explicate our notion of unsupervised patch-based context. In section 3, we

describe our approach for extracting the contextual prior and using it for improving performance at the classification tasks. Section 4 presents our results and demonstrates that useful contextual priors can indeed be extracted from unlabeled images.

2 Unsupervised Patch-based Context

Consider the image in Figure 1. A local region-based classifier does a good job at parsing this scene (see *Classifier Output* in figure). However as it does not reason about the high-level context of the scene, it makes mistakes in some of the regions (particularly those with unusual or confusing local patch-based features). Now let us consider we have access to the set of nearest neighbors to the query patch that all have the same underlying semantic configuration as the query but have different local patch-based features. Amongst the retrieved matches, lets assume the region-based classifier produces better results on some of them (compared to the result on the query). By marginalizing the outputs of the classifier on the retrieved matches, we can compute a useful prior probability over the region-classes for the local patch (referred to as *contextual prior*).

Two challenges arise:

- How can we retrieve the set of good nearest neighbor matches to a query patch (given the high intraclass variance and low interclass variance in local region features)?
- How can we ensure that the retrieved matches all share similar semantic characteristics as the query (but yet are not corrupted by the same mistakes as made on the query patch by the classifier)?

To address the first challenge, we consider using features not only from the query patch but also those extracted in a neighborhood around it while performing the matching step. The features arising in the neighborhood provide the much-needed *context* that helps in constraining the search and resolving ambiguities. The size of the neighborhood plays a crucial role. Many recent works have considered image-based context i.e., using the entire image as the neighborhood [15, 13, 14, 16]. Although global matching works well for some scenes (alley, shores), it is not possible to find good matches for all other types of scenes (city, outdoor neighborhood). To circumvent this issue, in this paper, we consider using sub-image neighborhoods for matching [17–19].

In order to retrieve matches that share the same underlying semantics as the query, we use the outputs of a supervised classifier (trained on a small set of labeled images) as semantic features for performing the matching step. As the classifier is trained to perform a specific task at hand, using its outputs on the local patch as well as in its neighborhoods helps in further constraining the search to the underlying task being solved. One potential problem with this method is that the supervised classifier would make similar mistakes on similar image regions, and thereby relying on those outputs as our features would retrieve matches that are corrupted by the same errors. This would result in computing

non-informative priors as marginalizing over the corrupted predictions would reinforce the mistakes and thus lead to no new information. To circumvent this problem, we rely only on the 'confident' outputs/predictions of the classifier while performing the matching step. In most general scenarios, the *easy* regions within an image are often confidently labeled by a supervised classifier. For a classifier exhibiting a low recall-high precision characteristic, its highly confident predictions are mostly correct and thus can be treated as weak form of ground-truth labels. Therefore by avoiding the non-confident regions and relying only on the confident predictions to guide the search process, we avoid retrieving matches that share the similar mistake patterns.

3 Approach

Our overall approach is as follows: Given a set of (few) labeled and (many) unlabeled images, we first train a supervised classifier (section 3.1) using a subset of labeled images as training data. We then run this classifier over the entire set of images (both labeled and unlabeled) to compute the *semantic* features over them. The semantic features are used (along with appearance features) to search the unlabeled images for retrieving nearest neighbor matches to every image patch in the labeled dataset (section 3.2). The retrieved matches are used to compute the contextual prior, which is subsequently used to update the supervised classifier so as to improve its performance (section 3.3).

3.1 Supervised Classifier

We use a multiple segmentation approach [2] to train our supervised classifier. Given an image and its corresponding superpixel map, simple features based on color, texture and location are extracted from the superpixel regions and are used to train a superpixel-similarity classifier. This classifier is used to group similar superpixels together to form larger segments. The larger segments offer better spatial support for extracting more complex region-based cues such as vanishing lines (geometry), shape, and boundary characteristics. These high-level features (in combination with the low-level cues) are used to train a region classifier so as to learn the mapping between the regions and their corresponding classes. Multiple segmentations are generated and the predictions are marginalized across the segments to assign final label confidences to each superpixel (i.e., probability p of a super-pixel belonging to one of the classes).

Merits and Limitations: The multiple segmentation approach is simple, fast yet powerful and has shown good performance at various tasks [20, 21, 2]. In our experiments too, we found that it achieves a good level of performance given limited amount of training data and is on a par with other state-of-the-art methods evaluated on the two selected datasets (see section 4).

The multiple-segmentation process is based on the hypothesis that some of the generated segments (in the soup of segments) would offer good spatial

support that is crucial for classification. Thus it must be noted that the process encourages homogeneous segments i.e., *local* regions belonging to a single class and does not encode higher-order contextual interactions/relations across classes (in fact such non-homogeneous segments are discouraged in this framework). This is exactly the knowledge we want to augment it with using the contextual prior.

3.2 Sub-Image Contextual Matching

Given an image (either labeled or unlabeled), we first divide it into non-overlapping 20×20 pixel patches yielding an $m \times n$ resolution grid (typically 15×20 for a 300×400 resized image). We extract two types of cues for the image at this resolution. For appearance, we use the GIST feature descriptor [22] which has been shown to perform well at grouping similar scenes [16]. We create this descriptor for each image at the $m \times n$ spatial resolution where each bin contains that image patch's average response to steerable filters at 8 orientations and 4 scales. For semantic context, we run the supervised classifier on the original image and discretize its output confidences to the resolution of the grid (hereby referred to as *semantic* feature). We mask out the semantic features in the *non-confident* regions and only use the features from the *confident* regions. A feature is *confident* if its max prediction value is above a precomputed threshold for the predicted class. The thresholds are decided based on the classifier's performance on a validation dataset.

Now the feature descriptor for a given patch in the image is built by concatenating the gist and the semantic features from a $k \times k$ square neighborhood around it. (In section 4, we provide details about the specific neighborhoods chosen). Using this feature descriptor, we compute distances for every patch in the query image to all the patches in the database images. We use L1 metric for computing the distances separately for the gist and the semantic features (We update the metric such that the norm in the unconfident regions for the semantic features is ignored). We scale the distances so that their standard deviations are roughly the same so that their influence in ordering the matches is equal. After sorting the aggregate feature distances, we pick the top K-nearest neighbors from amongst them. Rather than searching all the patches within all the 6 million unlabeled images for every query image patch, to circumvent the huge computational cost, we first retrieve the top 10000 global scene matches to the query using the approach of [16] and perform the costlier sub-image search within them. This entire process roughly takes 2 hours per query image (using unoptimized Matlab code) on a contemporary Xeon processor[1]. Our experiments are performed on a cluster of 400 processors. Some sample matches retrieved for query patches are shown in figure 1,2.

[1] the computational cost could be reduced by using several systems-level optimizations recently proposed in computer vision (e.g., branch-and-bound [23], hashing [24, 25] etc)

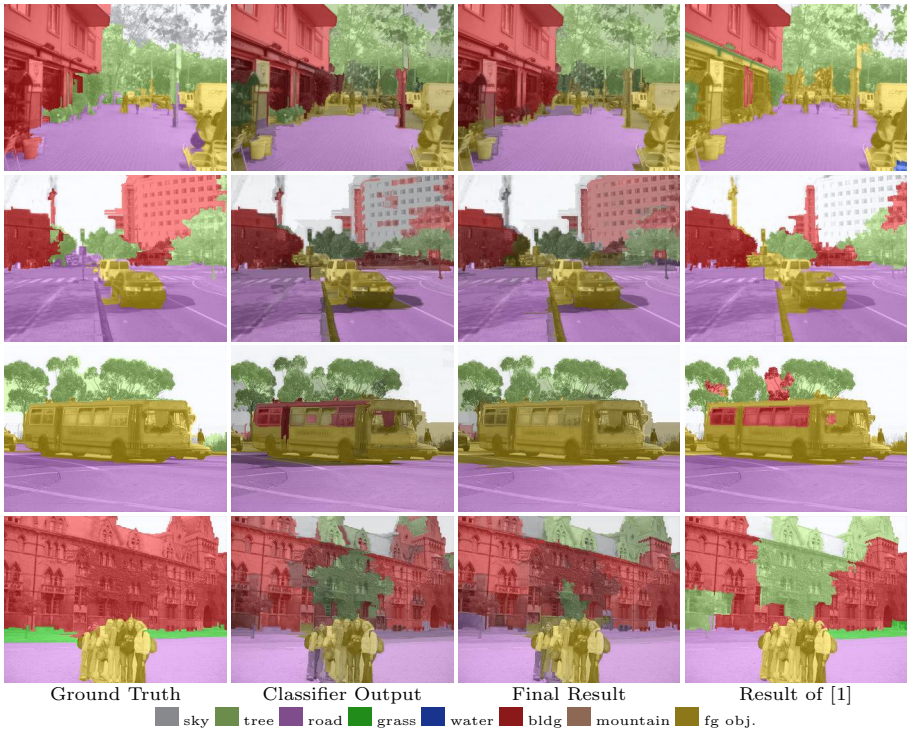

Fig. 2. Results on the region classification task. First and second row: part of the 'building' is misclassified as 'foreground'. Third row: parts of 'foreground' (bus) are labeled as 'building'. Last row: parts of 'building' misclassified as 'tree'. By retraining the classifier using the contextual prior, the mistakes have been rectified (Final Result). Results from [1] are also included for comparison.

3.3 Contextual Prior: Estimation and Usage

Given the top K-nearest neighbors $N_{1:K}$ retrieved for a query patch q in an image, its contextual prior P_q is computed by marginalizing the outputs of the supervised classifier $p(.)$ on the retrieved unlabeled matches i.e., $P_q = \sum_{i=1}^{K} p(N_i)$. As the matching process implicitly enforced the constraint to retrieve neighbors possessing similar underlying label characteristics as that of the query, by marginalizing their outputs, a good prior over the image classes is derived. This idea of encouraging similar scenes to have similar semantic labels, can be viewed as a weak form of manifold regularization [26].

In our experiments, we considered two methods for performing the marginalization step: a) direct marginalization of the classifier's outputs across the retrieved matches; b) marginalizing the outputs on the matches only when they are 'confident' (i.e., using N_i only when $p(N_i)$ is above a threshold). We empirically found that using around 15 matches in the first scheme and about 50 matches in the second scheme to perform equally well. We used the first scheme in our experiments.

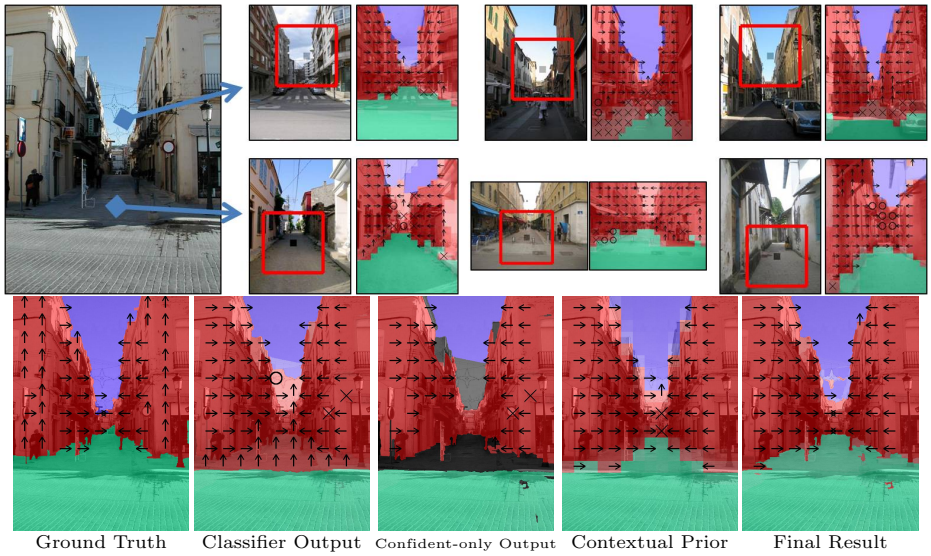

Fig. 3. Result on Geometric context dataset: Notice that the incorrectly classified ground and sky regions are corrected after the incorporation of the prior.

The estimated contextual prior acts as a useful cue for classification of the input image. In our experiments, we use it as an additional feature alongside the original set of features to retrain the supervised classifier as in [15]. In order to eliminate the variance in the results due to the randomness in the multiple segmentation process, we maintain the same segmentations as generated in the original training process. However it is indeed possible to use these features in the multiple segmentation process too (i.e., to retrain the superpixel-similarity classifier), which would further help to generate better segments/segmentations.

4 Results

We analyze the performance of our approach on the challenging region classification [1] and the surface-layout estimation [2] tasks. The unlabeled images used in our work are a collection of 6.5 million images downloaded from Flickr.

4.1 Region Classification

The region classification task is to classify the different regions within an image into one of the eight categories: *sky, grass, road, water, mountain, tree, building and foreground*. In [1], a unified region-based model that combined appearance and scene geometry to automatically decompose a scene into semantically meaningful regions was used. To compare the results obtained by our supervised learner to the one used in [1], we repeat the experiment following a similar set-up

	sky	tree	road	grass	water	bldg	mntn	fgob
sky	92.6	2.1	0.1	0.0	0.2	4.3	0.0	0.6
tree	4.9	61.4	1.2	0.9	0.1	26.6	0.1	4.9
road	0.2	1.4	88.0	0.5	1.0	4.0	0.0	4.9
grass	1.4	10.4	6.2	74.6	0.8	2.5	0.6	3.5
water	7.7	0.8	25.5	4.6	50.9	5.1	1.9	3.5
bldg	1.9	5.1	2.5	0.6	0.1	83.2	0.0	6.6
mntn	25.6	10.0	9.0	0.8	6.1	42.0	0.8	5.7
fgob	2.2	4.3	16.2	1.5	1.1	24.5	0.0	50.2

Baseline (Supervised) classifier

	sky	tree	road	grass	water	bldg	mntn	fgob
sky	93.2	2.9	0.0	0.0	0.4	2.9	0.0	0.6
tree	4.6	66.5	1.2	2.1	0.0	20.0	0.4	5.1
road	0.1	0.5	89.1	0.4	0.9	2.9	0.0	6.2
grass	0.5	6.0	2.7	84.0	1.8	0.6	0.1	4.3
water	7.1	0.3	24.5	5.6	50.3	2.9	1.3	8.0
bldg	1.7	6.6	2.3	1.2	0.1	81.3	0.1	6.8
mntn	26.2	20.1	8.4	2.0	6.1	12.0	3.0	22.3
fgob	2.8	4.3	14.0	2.0	1.0	17.5	0.1	58.3

After re-training with Contextual Prior

Table 1. Confusion matrix (row-normalized) of the supervised classifier before and after incorporating the contextual prior.

	grnd	vert	sky
grnd	83.0	16.6	0.3
vert	9.3	88.6	2.1
sky	0.1	9.7	90.2

Main-class

	left	front	right	porous	solid
left	36.8	36.5	7.3	9.9	9.6
front	7.0	53.8	11.9	17.5	9.8
right	4.0	23.6	49.9	11.3	11.1
porous	2.3	8.5	3.0	80.1	6.1
solid	4.5	18.7	7.1	18.6	51.1

Sub-class

Baseline (Supervised) classifier

	grnd	vert	sky
grnd	83.5	16.0	0.5
vert	9.0	89.4	1.6
sky	0.6	6.4	93.0

Main-class

	left	front	right	porous	solid
left	41.7	25.8	7.8	10.7	14.0
front	5.4	56.8	10.6	14.3	12.9
right	2.8	25.8	47.5	11.9	12.1
porous	1.3	6.6	2.1	83.2	6.8
solid	4.4	17.7	5.1	18.5	54.4

Sub-class

After re-training with Contextual Prior

Table 2. Confusion matrix (row-normalized) of the supervised classifier before and after incorporating the contextual prior.

as used in [1] but using the supervised learner as described in section 3.1. Quantitatively, we achieve a pixel-wise accuracy of 76.43%, which is similar to the result reported in [1] (76.4%). This confirms our baseline learner's performance as being on a par with the existing state-of-the-art on this dataset.

For our experiments with the unlabeled data, we divide the dataset of 715 images into 4 random splits - 100 images for training the superpixel-similarity classifier, 350 images for training the region classifier, 65 images for validation[2] and 200 images for testing. We train the supervised classifier using the 350+100 images and test it on the 200 test set. The pixel-based accuracy on the testset using this classifier is 75.6%. To obtain the *semantic* features on the training images (required for the matching step), we train/test a separate classifier on the training images using cross-validation.

Given a query image, we repeat the process described in section 3.2 to retrieve the nearest neighbors and to compute the contextual prior for retraining the classifiers. The updated classifier improves the result on the test set by 2.4% i.e.,

[2] The validation imageset is used to compute the confidence thresholds. They are set so as to achieve a minimum precision value (0.15 for mountains and 0.9 for remaining classes).

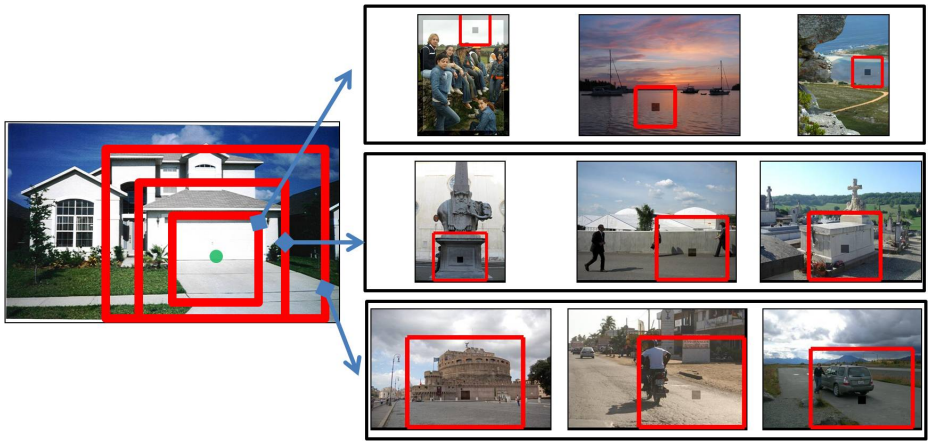

Fig. 4. Role of the context neighborhood size: Top few matches retrieved for a selected query image (with 15×20 resolution grid) patch using gist features at $5 \times 5, 9 \times 9$ and 13×13 neighborhoods from a set of 60000 unlabeled images. Having too small a neighborhood around the selected patch (green circle) leads to the top few matches being poor i.e., random matches on sky and seas as the region with the 5×5 sub-image does not have enough spatial context (thus the 'good' matches are lost in the potentially infinite matches). A larger neighborhood helps in retrieving better matches.

from 75.6% to 78.0%. The confusion matrix is shown in table 1. Although the increase in result seems small (in absolute numbers), it must be emphasized that the improvements made are significant. More specifically, our approach helps in correcting the mistakes made in classifying pedestrians/cars, tree branches or parts of buildings that typically occupy fewer percentage of pixels in the image (compared to sky and ground) but are crucial for successful image parsing. Qualitative results for some of the test images are displayed in figure 2,5. Observe that in the regions where the supervised classifier is unconfident (or incorrect) in its result, the contextual prior from unlabeled images helps in predicting the correct result.

4.2 Surface Layout Estimation

The goal of this task is to segment an image into meaningful geometric surfaces: *ground, planar-left, planar-frontal, planar-right, non-planar porous, non-planar solid, and sky.* In [2], an approach based on multiple segmentations was used on a dataset of 300 images. 50 images were used for training the superpixel-similarity classifier and the remaining 250 images were used for training/testing the region classifier in a 5-fold cross validation setup. We use the same splits and setup in our experiments. The accuracy obtained in our experiments was 87.1% for the main class and 59.3% for the sub-class. This is slightly different from (lower than) the one reported in [2]. We attribute the difference to the randomness in the segment generation process (which affects the results by +/- 1% as reported in [2]). However we use the same set of segmentations in the retraining process (with the contextual prior), so as to eliminate the variation in our subsequent results (due to the randomness).

Due to severe paucity of labeled data, we could not maintain separate train-val-test splits in this experiment. We used a classifier trained on the entire set of 250 images to run on all the unlabeled images. The thresholds were set based on the results obtained in the 5-fold cross validation process (to have a minimum precision of 0.9 for sky and ground, and 0.7 for the rest of the classes). Given a query image, we repeat the process described in section 3.2 to retrieve the nearest neighbors and to compute the contextual prior for retraining the classifiers. Quantitatively, for this task, we improve the results by 1.2% on the main-class (i.e., from 87.2% to 88.4%) and by 2.6% on sub-class (i.e., 59.3% to 61.9%). The confusion matrix is shown in table 2. The qualitative results are shown in Figure 3,6.

What is the right neighborhood for matching? The size of the neighborhood used for extracting features around a patch in the sub-image matching approach plays a non-trivial role (See figure 4). Having too small a neighborhood would lead to potentially many matches (as a lot of things get mapped to it and the good ones are lost amongst them), whereas using a global neighborhood (i.e., the entire image) would lead to too few or no matches. Indeed choosing the right size is data and task dependent. We experimented with various sizes of the neighborhood - 5×5, 9×9 and 13×13 for gist and 9×9 and 13×13 for the semantic features. We found the matches retrieved using a 9×9 neighborhood for gist and 13×13 neighborhood for semantic features to be good (based on the performance on validation set). We used these settings in all our experiments.

Standard Semi-Supervised Learning comparison. To compare the performance of our approach to a standard semi-supervised learning (SSL) algorithm that just takes labeled and unlabeled data together (with no intermediate labeling and using same data neighborhoods), we experimented with the multi-view SSL described in [27]. We trained classifiers using the available labeled data for various splits of our feature set and then applied them to all the unlabeled images for bootstrapping the initial classifiers with informative patches mined from them (i.e., patches that are classified with high confidence by at least one view but not all). This method failed to achieve any performance gains in our experiments. Due to the high appearance ambiguity of local patches across multiple feature views (e.g., a local patch of blue in a scene close to the horizon could either be 'sky' or 'water' unless a neighborhood around it is revealed), this method failed to gather informative samples. As a result, no new information is leveraged from the unlabeled images leading to no improvements in accuracy.

Finally in figure. 7, we show example results comparing our sub-image matching to other matching approaches. In order to support our hypothesis that sub-image matching helps retrieve improved matches over a global methods, we repeated our experiments by using the prior computed from global matches. More specifically, for each query image, we retrieve the top 50 global scene matches and compute the contextual prior by marginalizing the classifier outputs over the matches (on the entire image). For the region classification and the main-

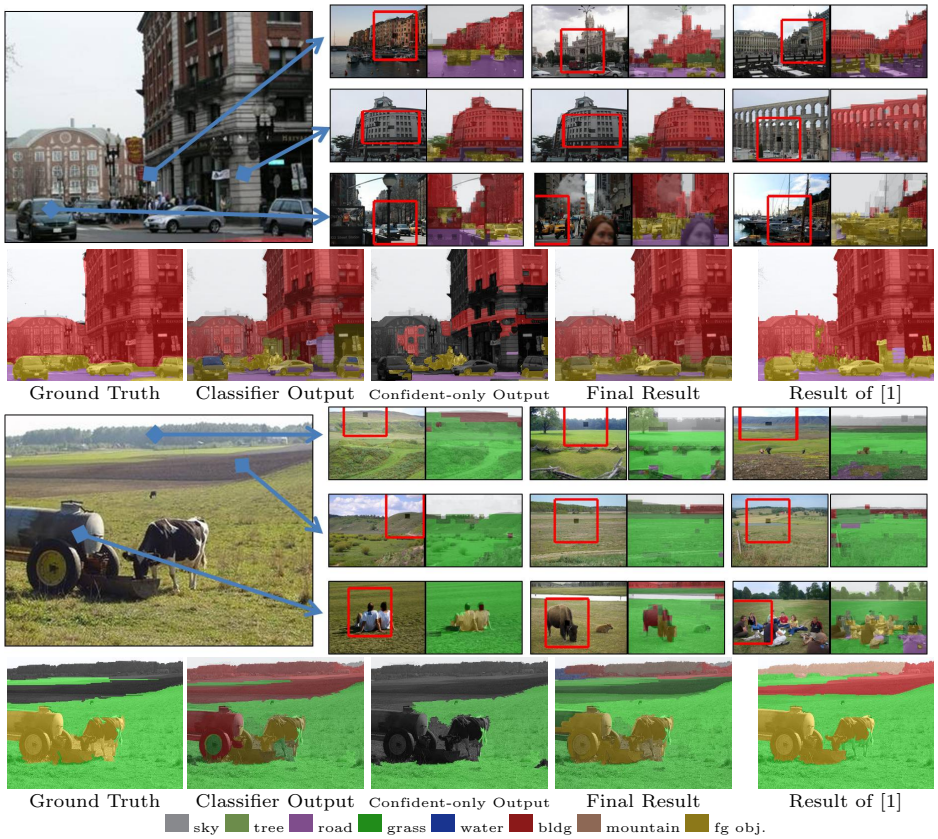

Fig. 5. Results on the region classification task. Top row: parts of the car and the building are misclassified. However the confidence of the correct prediction is increased by using the contextual prior from the retrieved matches. Bottom row: parts of the vehicle and grassland are incorrectly classified but corrected by the retrained classifier.

class surface layout estimation task, using this prior did not help in improving the result (the change in accuracies was less than 0.2%), while for the sub-class surface layout estimation task, the results improved by 2% (i.e., from 59.3% to 61.3%). Figure compares the matches retrieved using both the global and sub-image matching schemes for a few query images. Observe that the global matches get the gist of the scene right but do not localize the regions and the boundaries specific to a query patch, whereas using the sub-image approach retrieves much better matches. Further we also studied a semi-global way to obtain the matches. Instead of using a straight-forward L1 distance function over the entire image features, we weigh the distances using a Gaussian centered around the query patch so as to focus more on the distances in its immediate neighborhood while still matching weakly on the rest of the image. We found this method to retrieve good matches too. (However we chose the sub-image method as it performed equally well and is faster to compute.)

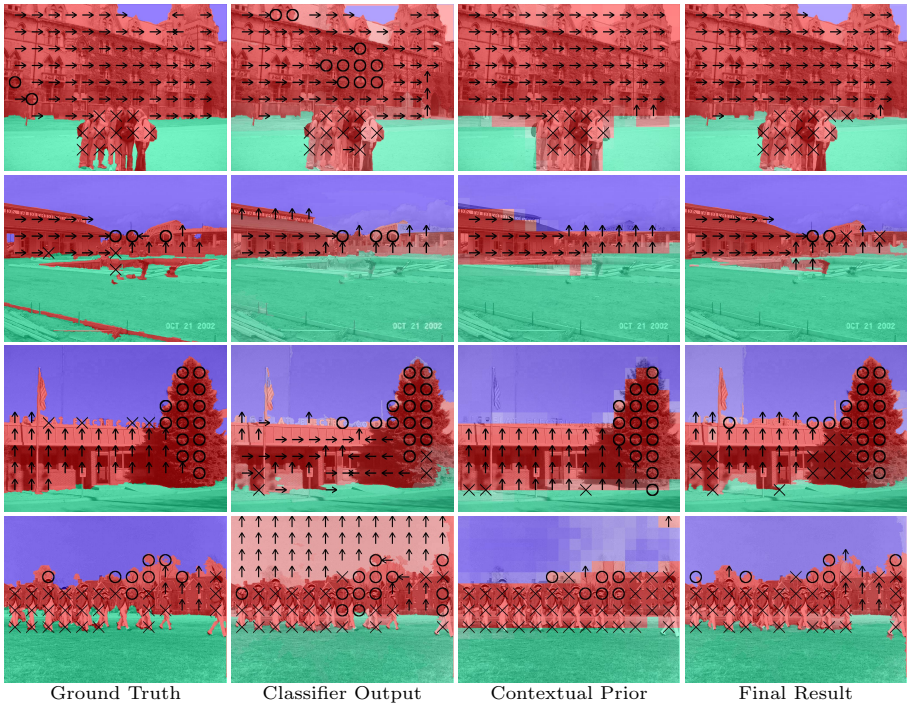

Fig. 6. Results on the geometric context dataset. ('X' indicates non-planar solid and 'O' indicates non-planar porous class). Top row: part of the left-facing building is misclassified as porous due to confusing texture. Second row: left-facing roof of the building is misclassified as 'frontal'. Third row: frontal face of the building is confused with 'left' and 'right' classes. Last row: sky is misclassified as vertical (frontal) class. In all cases, contextual prior computed from unlabeled images helps in improving the result.

5 Conclusion

Image interpretation is a hard problem as local evidence learned from a small set of labeled images is used for making scene-wise decisions. In this paper, we have presented an approach to alleviate the labeled data barrier by deriving contextual priors from completely unlabeled images for aiding supervised region-labeling classifiers. The main components of our approach are sub-image based matching, and semantic feature based similarity, that together enable us to encode *higher-order* context for measuring similarity and retrieving good matches. Beyond the region labeling tasks explored in this work, the proposed method allows us to leverage the huge collection of untapped Internet images in multiple interesting ways. For example, once the nearest neighbor matches to the query patches are retrieved, one could transfer any weak labels associated with the matches (e.g., Flickr tags or captions or any other annotations) onto the query and arrive at a competely data-driven interpretation of the query image.

References

1. Gould, S., Fulton, R., Koller, D.: Decomposing a scene into geometric and semantically consistent regions. In: ICCV. (2009)
2. Hoiem, D., Efros, A., Hebert., M.: Recovering surface layout from an image. IJCV **75** (2007)
3. Shotton, J., Winn, J.M., Rother, C., Criminisi, A.: Textonboost: Joint appearance, shape and context modeling for multi-class object recognition and segmentation. In: ECCV. (2006)
4. Li, L.J., Socher, R., Fei-Fei, L.: Towards total scene understanding:classification, annotation and segmentation in an automatic framework. In: CVPR. (2009)
5. Zhang, H., Xiao, J., Quan, L.: Supervised label transfer for semantic segmentation of street scenes. In: ECCV. (2010)
6. Zhu, X.: Semi-supervised learning literature survey. Technical Report 1530, University of Wisconsin Madison (2008)
7. Singh, A., Nowak, R., Zhu, X.: Unlabeled data: Now it helps, now it does not. In: NIPS. (2008)
8. Munoz, D., Bagnell, J.A., Hebert, M.: On two methods for semi-supervised structured prediction. Robotics Institute, Carnegie Mellon University (2010)
9. Tu, Z.: Auto-context and its application to high-level vision tasks. In: CVPR. (2008)
10. Munoz, D., Bagnell, J., Hebert, M.: Stacked hierarchical labeling. In: ECCV. (2010)
11. He, X., Zemel, R.S., Carreira-Perpiñán, M.Á.: Multiscale conditional random fields for image labeling. In: CVPR. (2004)
12. Ramalingam, S., Kohli, P., Alahari, K., Torr, P.: Exact inference in multi-label crfs with higher order cliques. In: CVPR. (2008)
13. Liu, C., Yuen, J., Torralba, A.: Nonparametric scene parsing: Label transfer via dense scene alignment. In: CVPR. (2009)
14. Russell, B., Torralba, A., Liu, C., Fergus, R., Freeman, W.T.: Object recognition by scene alignment. In: NIPS. (2007)
15. Divvala, S.K., Efros, A., Hebert, M.: Can similar scenes help surface layout estimation? CVPR 2008, IEEE Workshop on Internet Vision (2008)
16. Hays, J., Efros, A.A.: Scene completion using millions of photographs. SIGGRAPH **26** (2007)
17. Russell, B.C., Efros, A.A., Sivic, J., Freeman, W.T., Zisserman, A.: Segmenting scenes by matching image composites. In: NIPS. (2009)
18. Tighe, J., Lazebnik, S.: Superparsing: Scalable nonparametric image parsing with superpixels. In: ECCV. (2010)
19. Efros, A.A., Leung, T.K.: Texture synthesis by non-parametric sampling. In: Proc. ICCV, Corfu, Greece (1999) 1033–1038
20. Russell, B.C., Efros, A.A., Sivic, J., Freeman, W.T., Zisserman, A.: Using multiple segmentations to discover objects and their extent in image collections. In: Proc. CVPR. (2006)
21. Malisiewicz, T., Efros, A.A.: Improving spatial support for objects via multiple segmentations. In: BMVC. (2007)
22. Oliva, A., Torralba, A.: Building the gist of a scene: the role of global image features in recognition. Progress in brain research **155** (2006) 23–36
23. Lampert, C.H.: Detecting objects in large image collections and videos by efficient subimage retrieval. In: ICCV. (2009)
24. Sivic, J., Zisserman, A.: Efficient visual search of videos cast as text retrieval. In: PAMI. (2009)
25. Jain, P., Vijayanarasimhan, S., Grauman, K.: Hashing hyperplane queries to near points with applications to large-scale active learning. In: NIPS. (2010)
26. Belkin, M., Niyogi, P., Sindhwani, V.: On manifold regularization. In: AISTAT. (2005)
27. Blum, A., Mitchell, T.: Combining labeled and unlabeled data with co-training. In: COLT. (1998)

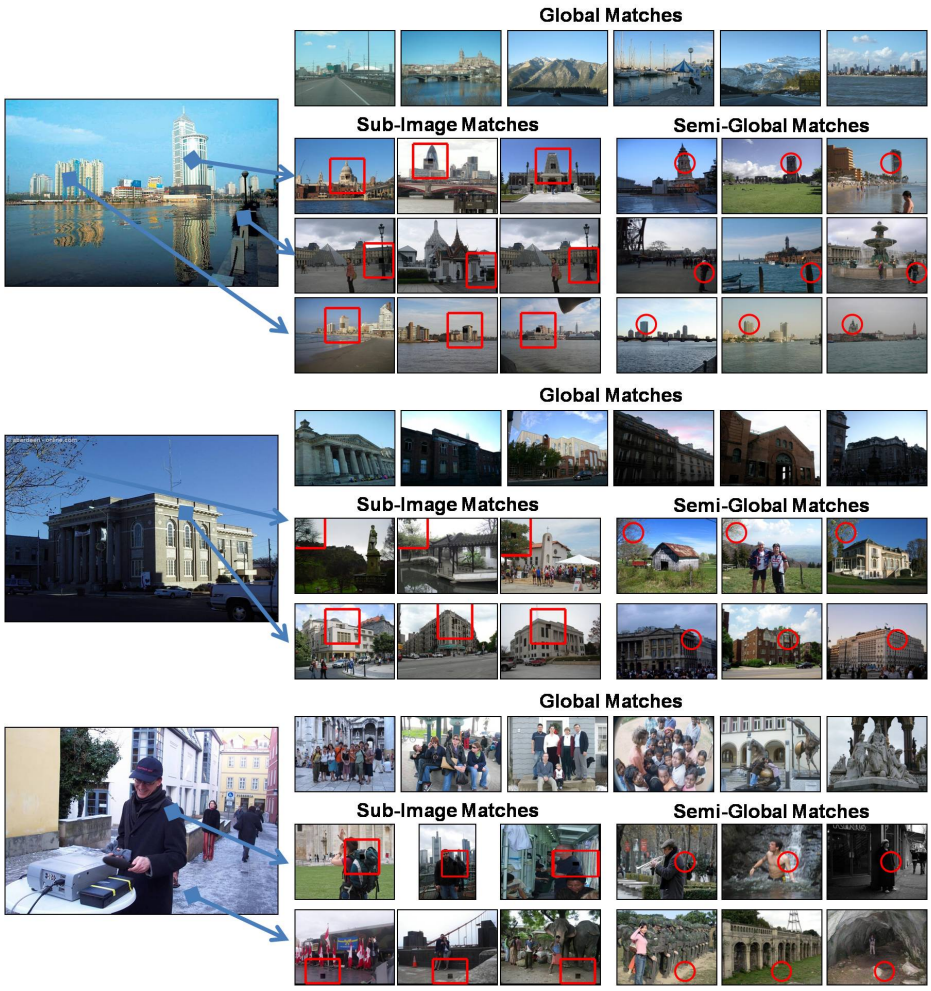

Fig. 7. Global vs Sub-image matching: The matches retrieved using features from the entire image do well in getting the overall gist of the scene but fail to match the individual regions within the image. By using the sub-image based approach, we retrieve better matches. The semi-global approach is also displayed for comparison.