Aspect Coherence for Graph-Based Semantic Image Labelling

Giuseppe Passino Ioannis Patras

Ebroul Izquierdo

Queen Mary, University of London

Mile End Road, London, E1 4NS, UK

{giuseppe.passino,ioannis.patras,ebroul.izquierdo}@elec.qmul.ac.uk

May 25, 2009

Abstract

Image semantic segmentation is the task of associating a semantic category label to each image pixel. This classification problem is characterised by pixel dependencies at different scales. On a small-scale pixel correlation is related to object instance sharing, while on a middle- and large-scale to category co-presence and relative location constraints. The contribution of this paper is twofold. First, we present a semantic segmentation framework that jointly learns category appearances and small- and middle-scale pixel dependencies. The algorithm computational complexity is reduced by considering these two classes of dependencies separately. In particular, small-scale dependencies are accounted by clustering pixels into larger patches via image oversegmentation. To tackle middle-scale dependencies we propose a system based on a Conditional Random Field (CRF), that is, a discriminative graphical model, built over the patches. In this context, we propose a novel strategy to exploit local patch aspect coherence to impose an optimised structure in the graph that allows to have exact and efficient inference. The second contribution is a method

to account for full patch neighbourhoods without resorting to graphical structures containing loops. We introduce the concept of *weak neighbours*, which are connected to a patch in the image but not in the chosen graph. They are pre-classified according to their visual appearance and their category distribution probability is then used in the CRF inference step. We present experimental evidence of the validity of the method, showing improvements in comparison to other works in the field.

1 Introduction

Image understanding is the process of recognising and establishing relations between objects depicted in a scene. The tasks of object detection, localisation, image classification and image semantic segmentation are strongly related to understanding. In particular, semantic segmentation, or semantic labelling, is the task of associating a category label to each pixel of an image, obtaining a category label map as a result. An example of such a segmentation is presented in Fig. 1 for an image from the Microsoft Cambridge (MSRC) image dataset[1], used throughout the paper and for the experimental evaluation. The key difference of the semantic segmentation task with the one of object detection is that in the first case there is no concept of object instances. On the contrary, object detection algorithms often model object instances as structured entities. Therefore, the latter ones deal only with an object category at time and fail contextualising the result in a global scene classification. They also are intrinsically less scalable to many categories, due to the fact that a separate, complex, structured model has to be present for each object category. Semantic segmentation has applications in many areas, including human-computer interaction, image retrieval, and automated systems.

In low-level semantic segmentation, image pixels are analysed and classified according to features locally extracted. However, it is of great importance to consider contextualisation during the classification. For example, adjacent

[1] Available on-line: `http://research.microsoft.com/vision/cambridge/recognition/`.

pixels have similar properties and have high probability to belong to the same object. There is therefore a strong short-scale correlation between them. Additionally, there are middle- and large-scale relationships between categories to be considered: for instance, co-presence and neighbourhood probabilities strongly depend on categories.

System overview and contributions. We propose a part-based approach for image semantic segmentation, presenting two main contributions in the area. The first contribution is a full semantic segmentation framework that jointly models the appearance of different categories and the relationships between these categories. A diagram of the system is represented in Fig. 2. In particular, with our method the complexity of modelling dependencies at different scales is broken by a two-tier approach. In a first step, homogeneous patches are extracted from the images. Pixels are grouped into coarser patches, or *super-pixels*, that are homogeneous in terms of features and therefore highly likely to be part of a single object instance. This step addresses the short-scale pixel dependencies. In the second step, the image is analysed at patch-level, with the advantage of having simpler structures to model and classify, without losing precision on region boundaries. Patch appearance and dependencies are modelled via a Conditional Random Field (CRF), that is, an undirected discriminative probabilistic graphical model, that is built over the patches and describes the probability of different patch labelling configurations. The CRF model accounts for middle-scale dependencies. Patches are represented in terms of colour- and texture-based features locally extracted. These features represent the observation in the labelling model and are also used as a support information for the graphical structure generation. The structural choice is important because considering all the dependencies between different patches can make the problem intractable. For this reason, we propose a method to generate an effective loop-less graph in which inference can be performed efficiently. To this end, we build a tree over the patches encouraging connections between patches

that are coherent in appearance, thus maximising the expected local correlation. The second contribution presented in the paper is related to the previously described labelling framework, and in particular to the structural choice. Basing the analysis on a tree limits the amount of considered local context for patches, since many neighbours are not connected. We propose a method to partially account for full patch neighbourhoods, without introducing loops in the graph. This is based on a two steps classification and the consideration of weak priors for neighbours. Patches are at first classified independently according to their appearance and then the obtained distribution over the category labels is used as additional feature when considering neighbours relationships for nodes not connected in the tree.

The paper is organised as follows: Sec. 2 presents a brief review of works related to the semantic labelling problem, outlining differences with our proposal. In Sec. 3 the segmentation algorithm used to obtain the image patches is discussed, while the features associated to the obtained patches are discussed in Sec. 4. The learning process is treated in Sec. 5, and Sec. 6 is devoted to the training and inference in the proposed model. Experimental results and a comparison with other works in the area are presented in Sec. 7. Finally, Sec. 8 comments on the proposed approach, briefly discussing possible future directions of work.

2 Related Work

One of the most popular and successful methods in associating categories to features in images is the probabilistic Latent Semantic Analysis (pLSA) [1], an application to the image domain of the bag-of-words framework [2]. This method is indeed based on *latent* topics, therefore a pixel level labelled ground truth is not required for training. Traditionally, features are extracted at salient points and then clustered into visual words. To use pLSA for semantic segmentation, Verbeek and Triggs [3] extract visual words in a dense grid and take advantage

of the labelling availability to impose constraints on the topics distributions during training. In their work, spatial consistency is enforced through a Markov Random Field (MRF), a generative probabilistic graphical model. However, in contrast with our proposal, their work is mainly based on the bag-of-words paradigm. In the basic formulation of pLSA spatial information related to visual words is ignored. This simplifies the approach but on the other hand strong correlations between close words are ignored as well. Recently, a number of approaches addressed this shortcoming [3, 4, 5].

Considering explicit pixels or patch dependencies in a probabilistic model is in general intractable for dimensionality reasons, resulting in a probability function of hundreds or thousands of variables. Probabilistic graphical models ease this task because they allow to explicitly specify direct dependencies between elements [6]. In particular, discriminative models have recently witnessed remarkable success in object detection and semantic segmentation tasks [7]. Conditional Random Fields (CRFs) [8] have proved to be a valuable tool for semantic labelling [9, 10, 11]. A CRF is the discriminative version of a MRF, an undirected probabilistic graphical model in which a configuration likelihood is described through a function factorisable on the graph cliques. To limit complexity, only local connections are usually considered. Therefore, CRFs are a useful way to account for short- and middle-scale dependencies. The field can be applied at pixel level [9, 12] or patch level [10, 11, 13].

For models applied at pixel-level, CRF will tend to enforce labelling coherence at a short-scale. An advantage of this approach is the possibility for an accurate object segmentation, but on the other side the size of the resulting graph makes the training step complex. Moreover, in order to model longer range dependencies, additional strategies are needed. *TextonBoost* and its extensions [9, 12] offer an example of pixel-based approach. Longer range interactions are considered by using a boosting approach on texture descriptors and clustering pixels in different regions at image level. The main drawbacks of such a method are the hard-thresholded decision in the texture-based clustering and

the high complexity associated with the training of the system. However the achievable precision in the object boundary detection is high.

A different approach, chosen in our work, is to apply the CRF at patch-level. Usually methods operating at patch-level obtain a lower precision in the category boundaries, due to the coarse scale labelling, in favour of a more lightweight system that additionally models less-local dependencies. Verbeek and Triggs [11] propose a CRF that is applied to a grid of rectangular patches extracted from the image. The method explicitly addresses the presence of multiple categories within a single patch. This event often occurs with their method because the patch extraction is not driven by the image content but is fixed. This drawback also introduces errors in the final labelling, where all the pixels assigned to a patch are labelled uniformly. Global and long range dependencies are considered in terms of global and distributed histograms of visual words. In this paper we use oversegmentation to improve category boundary detection, and to have more consistent patches as an input to the labelling block. An additional difference with the work of Verbeek and Triggs is that with the CRF we jointly model appearance and relationships between patches, not recurring to visual words that can discard important visual information when computed.

Furthermore, CRFs have been used to impose layout constraints only, without inter-patches local connections [10]. In this method the role played by the graph is fundamentally different. The CRF is used to impose layout configurations learnt from the dataset. The spatial patterns are used to influence labelling spatial configurations over the image. The use of a defined number of allowed layouts presents limited generalisation properties. Our pairwise inter-patches connections represent more generalisable relationships and favour smooth labelling.

For fields built over patches, these can be selected in order to ease the learning of a coherent appearance model for them. Rectangular patches are easy to extract but they present problems related to the presence of mixed categories within them. Another possibility is to obtain patches through oversegmen-

tation [14], as in this paper. Toyoda and Hasegawa [13] in particular use a colour-based segmentation approach. In general, this choice does not lead to robust and reliable patch boundaries, since colour alone is not a sufficient clue to separate object instances. In the aforementioned method, the authors use distributed features, as well as distributed categories compatibility tables, to consider long range dependencies. This makes the model very complex in terms of inference, and its parameters are simply imposed rather than be learnt in a training phase. He *et al.* [15] use the Normalised Cuts (NCuts) algorithm [16] to obtain oversegmentation. The resulting patches are analysed with a mixture of standard CRFs in which neighbouring patch labels dependencies are modelled through a categories compatibility table. In contrast to our work, the appearance for the single CRF is learnt separately using a multilayer perceptron classifier, leading to suboptimal performance.

3 From Pixels to Patches

In this work we use the NCuts algorithm [16], similarly to He *et al.* [15], to obtain homogeneous patches. In this way we break the problem complexity by separately addressing very short-scale pixel correlation and middle-scale patches dependencies. Unsupervised segmentation is based on homogeneity of pixels clusters with respect to a certain metric. With the choice of the correct metric, based on both colour and texture information, oversegmentation is a valid strategy to isolate groups of pixels that are very likely to belong to a single instance of an object. Therefore, we combine the advantages of patch-based approaches with a generally accurate segmentation result. An accurate segmentation is fundamental since errors at this stage can not be recovered in the labelling phase. The NCuts algorithm represents the state of the art in image segmentation, even though its requirements in terms of time and memory are substantial. However, in contexts where resources are limited, other segmentation approaches can be used instead, since the segmentation scenario is not critical in this case due to

the high number of target patches.

The NCuts algorithm is a spectral clustering method that aims at grouping connected pixels according to a similarity measure. To this end, we define the similarity matrix $\mathbf{W} = \{w_{ij}\}$ in which w_{ij} measures the similarity between the pixels i and j. Additionally, we consider pixels as nodes in a graph $G = (\mathcal{V}, \mathcal{E})$, in which edges in $\mathcal{E}$ are weighted according to $\mathbf{W}$. It is therefore possible to define the *cut* between the disjoint partitions $\mathcal{A}, \mathcal{B} \subseteq \mathcal{V}$ as

$$\mathrm{cut}(\mathcal{A}, \mathcal{B}) = \sum_{a \in \mathcal{A}, b \in \mathcal{B}} w_{ab} \ , \tag{1}$$

and the volume of $\mathcal{A}$ as

$$\mathrm{vol}(\mathcal{A}) = \sum_{a \in \mathcal{A}, v \in \mathcal{V}} w_{av} \ . \tag{2}$$

The NCuts algorithm for K clusters minimises the cost function

$$\mathrm{KNcuts}(\mathcal{V}_1, \ldots, \mathcal{V}_K) = \sum_{k=1}^{K} \frac{\mathrm{cut}(\mathcal{V}_k, \mathcal{V} \backslash \mathcal{V}_k)}{\mathrm{vol}(\mathcal{V}_k)} \ , \tag{3}$$

where $\mathcal{V}_i \cap \mathcal{V}_j = \emptyset$ for $i \neq j$ and $\bigcup_k \mathcal{V}_k = \mathcal{V}$. This problem can be solved efficiently (although not exactly) by computing the eigenvalues and eigenvectors of the generalised eigenproblem

$$(\mathbf{D} - \mathbf{W})\mathbf{y} = \lambda \mathbf{D}\mathbf{y} \ , \tag{4}$$

where $\mathbf{D}$ is the diagonal matrix of the vertices degrees $d_i = \sum_j w_{ij}$. Minimising Eq. (3) in practise leads to balanced regions that will have a comparable area due to the terms $\mathrm{vol}(\mathcal{V}_k)$.

To calculate $\mathbf{W}$ we use the similarity measure described by Martin *et al.* [17]. Region boundaries can occur either due to the presence of strong edges, or due to a change of the texture pattern. The nature of these two kind of boundaries is different. Boundaries due to edges present a strong response to gradient-based

features. On the other side, textures are well represented by the response to Gaussian filterbanks. The weight between two pixels is inversely proportional to the probability that an object boundary is present between them. This probability is obtained by first evaluating texture- and colour-based boundary presence probabilities separately and then fusing the result via a logistic regression. An example of the segmentation results obtained with this approach is given on the left of Fig. 3, together with the related ground truths for a visual evaluation of the accuracies of the categories boundaries.

4 Feature Extraction

Object instances of different categories can not be discriminated with the same degree of accuracy with the use of a single type of feature. Instead, multiple features related to different traits of the patches have to be obtained. A feature that is highly discriminative for one category can be almost uninformative for another one. Three types of features have been used:

Texture/edge features: texton descriptors, as described by Malik *et al.* [18], are extracted. Textons are histograms of visual words obtained clustering vectors computed at pixel level by applying oriented filterbanks at different scales. Unlike in Malik *et al.*, though, the visual words dictionary (consisting of 300 words) is obtained from the entire dataset and shared among images, rather than being image-based. For each patch, the histogram is extracted taking the entire area as a support, but weighting pixel contributions with a Gaussian window centred at the patch centre of gravity. Finally, the dimensionality of the descriptor is further reduced to 40 by Principal Component Analysis (PCA).

Colour features: we use the robust invariant hue descriptor introduced by van de Weijer and Schmid [19]. This is a histogram of hue values obtained from a normalised image weighted on the colour saturation values. We use histograms of 20 bins. The support for these features is the entire patch

area, again with the contribution of different pixels weighted based on the distance from the patch centre of gravity. A smoothing of the histogram reduces the impact of the quantisation error.

Position features: the normalised position of the patch centre of gravity is considered as additional feature, to account for the weak information associated to the location within the image of certain categories such as sky or grass.

5 Labelling Subsystem

To demonstrate the role of the aspect coherence in the patching process we start by considering a simple discriminative model that treats different patches as independent (no structural information). In a second phase, this model is enriched with neighbouring categories compatibility tables, obtaining a CRF model, in which connections have to be accurately chosen. Finally, we introduce a method to incorporate full neighbourhoods awareness at patch level without introducing loops in the graph, using a two stage labelling.

5.1 Independent Patches Discriminative Model

Initially, we aim at labelling an image by analysing patches independently. For each image, we extract a set of descriptors $\mathbf{X}$ composed by the feature vectors $\mathbf{x}_j \in \mathbb{R}^n$ associated to each patch j. We use a discriminative model to learn independently the appearance of each patch. A *softmax* function expresses the probability that the patch j takes the label $y \in \mathcal{L}$, given the observation vector $\mathbf{x}_j$ as

$$p(y|\mathbf{x}_j;\theta) = \frac{e^{\theta_y \cdot \mathbf{x}_j}}{\sum_{y' \in \mathcal{L}} e^{\theta_{y'} \cdot \mathbf{x}_j}} \ . \tag{5}$$

The model parameter vectors θ_y express the compatibility between the appearance vector $\mathbf{x}_j$ and the label y. The patches are independent, so the probability

of a labelling $\mathbf{y} = \{y_1, \ldots, y_m\}$ for the entire image is

$$p(\mathbf{y}|\mathbf{X};\theta) = \prod_{j=1}^{m} p(y_j|\mathbf{x}_j;\theta) \ , \tag{6}$$

where j spans over the m image patches.

5.2 CRF Model

Dependencies between neighbouring patches can be taken into account by extending the model presented in the previous section and adding to Eq. (5) factors of multiple variables, which results in a CRF. A CRF is defined over a graph $G = (\mathcal{V}, \mathcal{E})$. The node $v_j \in \mathcal{V}$ is related to the j-th patch category label variable $y_j \in \mathcal{Y}$. Edges in the graph represent direct probabilistic dependencies between these variables. The graph is Markovian, that is, each variable is independent on the entire graph when conditioned on its neighbours,

$$p(y_j|\mathcal{Y} \setminus \{y_j\}) = p(y_j|\mathcal{N}_{y_j}), \qquad \mathcal{N}_{y_j} = \{y_k : (j,k) \in \mathcal{E}\} \ . \tag{7}$$

Under this assumption, the CRF can express probabilities that are in the form of a Gibbs distribution

$$p(\mathbf{y}|\mathbf{X};\theta) = \frac{e^{\Psi(\mathbf{y},\mathbf{X};\theta)}}{Z(\mathbf{X};\theta)} \ , \tag{8}$$

where Z is a normalisation factor,

$$Z(\mathbf{X};\theta) = \sum_{\mathbf{y}\in\mathcal{L}^n} \exp\left(\Psi(\mathbf{y},\mathbf{X};\theta)\right) \ . \tag{9}$$

The so-called *local function* Ψ has the form

$$\Psi(\mathbf{y},\mathbf{X};\theta) = \sum_{c\in\mathcal{C}} \phi_c(\mathbf{y}_c,\mathbf{X};\theta) \ , \tag{10}$$

where $\mathcal{C}$ is the set of cliques in the graph, ϕ_c is the *potential function* associated to the clique c and $\mathbf{y}_c$ is the projection of $\mathbf{y}$ in c, that is, $\mathbf{y}_c = \{y_i : v_i \in c\}$. If

only singleton potential functions (one-node cliques) are considered, the model falls back into a softmax function. Instead, we now define singleton and pairwise potential functions, used to model patch appearance and neighbour relationships respectively. Therefore, the local function assumes the form

$$\Psi(\mathbf{y}, \mathbf{X}; \theta) = \sum_{v \in \mathcal{V}} \sum_{k \in \mathcal{K}_1} \theta_k \phi_k^1(y_v, \mathbf{X}) + \sum_{(i,j) \in \mathcal{E}} \sum_{k \in \mathcal{K}_2} \theta_k \phi_k^2(y_i, y_j, \mathbf{X}) \ , \qquad (11)$$

where in Eq. (11) we made the dependency of the local function on the parameter vector θ explicit by weighting each potential function by a factor θ_k; $\mathcal{K}_{1,2}$ are the set of indices k of the parameter vector θ referring to different unary and pairwise potentials.

Potential Functions. In Eq. (11), the singleton potential functions ϕ_k^1 encode the compatibility between feature vectors and category labels. They depend on local patch features, $\mathbf{x}_v$, and have the form of selector functions, $\phi_k^1(y_v, \mathbf{x}_v) = x_{vu_k} \delta(y_v, l_k)$, where $u_k \in [1, n]$ spans the feature vector and $l_k \in \mathcal{L}$ the sets of possible category labels. The function δ is the Kronecker's delta. The patch feature vectors $\mathbf{x}_v$ are obtained by combining the different descriptors detailed in Sec. 4, concatenating the relative feature vectors. The functions ψ_k^1 represent, in a more flexible notation that is required for the CRF treatment, the same log-linear model used in Eq. (5) for the independent patches model. Therefore, the equality $\theta_l \cdot \mathbf{x}_v = \sum_{k \in \mathcal{K}_{1,l}} \theta_k \phi_k^1(y_v, \mathbf{x}_v)$ holds, where $\mathcal{K}_{1,l} \subset \mathcal{K}_1$ contains all the indices k for which $l_k = l$.

The pairwise potential functions implement a compatibility Look-Up Table (LUT) between category labels, and they are selector functions $\phi_k^2(y_i, y_j) = \delta(y_i, l_k)\delta(y_j, l'_k)$. There is no dependency from the observation related to the two patches, and the functions are therefore symmetrical. Compatibilities are learnt in terms of magnitude of the coefficients θ_k associated to each function. An alternate choice of pairwise functions considers the difference of appearance for the pair of patches. We also considered in the experiments the choice of

potential functions $\phi_k^2(y_i, y_j, \mathbf{x}'_i, \mathbf{x}'_j) = (x'_{iu_k} - x'_{ju_k})\delta(y_i, l_k)\delta(y_j, l'_k)$, that are further weighted on the u_k element of the difference between the feature vectors $\mathbf{x}'$, that in our case are the hue part of the histogram.

Graph Connections and Aspect Coherence. We opt for a tree structure to have exact and efficient inference during training (as detailed in Sec. 6). The method proposed in this paper is to exploit information on the aspect of the patches in the tree-choice phase. Starting from an initial graph in which all the nodes corresponding to neighbouring patches are connected, a tree is obtained using the Minimum Spanning Tree (MST) algorithm. Connections between patches that are coherent in aspect are encouraged by weighting graph edges on appearance similarity. The distance between patch colour features has been chosen as edge weight. The colour feature is used for different reasons. First, sharp colour changes are often a clear indication of an object boundary. Even if colour itself is not a good descriptor for some categories, other authors [9] have noticed that it tends to be shared within object instances. Additionally, the histogram form of the robust hue descriptor detailed in Sec. 4 offers a suitable support for the use of a consistent distance metric. Finally, the hue descriptor equally describes all the patches, except in rare cases of limited illumination in which the hue measure is not reliable. This is in contrast with texture features, that poorly describe instances of some categories (*e.g.* clear sky, some buildings, car frames). The metric used to calculate the distance between colour feature vectors is the symmetric Kullback-Leibler divergence, defined for two distributions P, Q as

$$D_{KLs}(P||Q) = D_{KL}(P||Q) + D_{KL}(Q||P) \ , \tag{12}$$

where D_{KL} is the (asymmetric) Kullback-Leibler divergence

$$D_{KL}(P||Q) = \sum_i P(i) \log \frac{P(i)}{Q(i)} \ . \tag{13}$$

Results of the MST algorithm are presented in Fig. 3: it is possible to observe how different objects are very little connected, most of the edges lying between patches of the same category.

Weak Neighbourhood. We introduce *weak neighbours* to account for full neighbourhoods while labelling patches without introducing loops in the graph. Given a segmented image and a graph built over its patches, the weak neighbours of a patch are all the adjacent patches not linked in the tree. When performing the classification using CRF, normal neighbour connections are modelled as in Eq. (11). Weak neighbours contribute to single node potentials by the means of previously computed category distributions integrated as additional features. The estimation of the probability distribution over the category labels for each patch is obtained by pre-classifying it according to the independent patches model presented in Sec. 5.1. The reason why a neighbour is "weak" for a given patch is that the distribution used in the interaction does not change during inference on CRF. In this way circular interactions are avoided (the only interaction between two nodes during the inference takes place via the path connecting them in the tree).

The feature vector containing the category distribution for a weak neighbour of a patch v can be introduced in the local function in Eq. (11) with single node potentials of the form $\phi_k^{1,w}(y_v, \mathbf{p}) = p_{u_k}\delta(y_v, l_k)$, where $\mathbf{p}$ is the distribution over category labels of that weak neighbour of v. The local function will then assume the form

$$\begin{aligned}\Psi(\mathbf{y}, \mathbf{X}; \theta) = &\sum_{v\in\mathcal{V}}\sum_{k\in\mathcal{K}_1}\theta_k\phi_k^1(y_v, \mathbf{x}_v) + \sum_{v\in\mathcal{V}}\sum_{j\in\mathcal{N}_v^w}\sum_{k\in\mathcal{K}_1^w}\theta_k\phi_k^{1,w}(y_v, \mathbf{p}_j) + \\ &+ \sum_{(i,j)\in\mathcal{E}}\sum_{k\in\mathcal{K}_2}\theta_k\phi_k^2(y_i, y_j, \mathbf{X}) \ , \end{aligned} \tag{14}$$

where $\mathcal{N}_v^w$ indicates the set of weak neighbours for the node v and $\mathcal{K}_1^w$ the corresponding parameter vector indices. In Fig. 4 an example of the graph used for the CRF is presented, in which dashed lines represent weak neighbour

connections.

6 Training and Inference

Training and inference are performed by maximising the Maximum A Posteriori (MAP) probability for the training set labelling. In the independent patches model the log-likelihood used during training is

$$L(\theta) = \sum_{i=1}^{N} \log(p(\mathbf{y}_{ai}|\mathbf{X}_i;\theta)) = \sum_{i=1}^{N}\sum_{j=1}^{m_{ai}} \log(p(y_{aij}|\mathbf{x}_{ij};\theta)) \tag{15}$$

where the index i spans over all the N images in the training set, m_{ai} is the number of labelled patches for the i-th image and $\mathbf{y}_{ai}$ is the corresponding ground-truth labelling. As ground truth label for a patch we considered the label of the majority of the patch pixels. Eq. (15) can not be optimised on θ in closed form but a gradient ascent iterative optimisation method is used instead. In this work, we employ the L-BFGS algorithm [20] for its fast convergence properties. The choice of the starting point for the L-BFGS algorithm does not influence the results, since the optimisation problem has a single global maximum (as when connections are considered, in the CRF model). The k-th gradient component of Eq. (15) is

$$\frac{\delta L}{\delta \theta_k}(\theta) = \sum_{i=1}^{N}\sum_{j=1}^{m_{ai}} x_{iju_k}(\delta(y_{aij}, l_k) - p(l_k|\mathbf{x}_{ij};\theta)) \ , \tag{16}$$

where u_k and l_k are the feature vector index and category associated to the k-th parameter vector coefficient (to be consistent with Eq. (5) we consider k spanning over all the coefficients of the parameter vectors associated to each category). Since the patches are considered and modelled as independent, the ones that are unlabelled in the training set can be ignored. Inference is performed

choosing, for each image patch j, the category

$$y_{j,\text{opt}} = \arg\max_{l}\{\theta_l \cdot \mathbf{x}_j\} \ . \tag{17}$$

In the CRF model the patches are no more independent. However, as long as the graph presents no loops, efficient and exact inference can be performed via Belief Propagation (BP). Unlabelled patches cannot be ignored as in the independent patches model, because they contribute to the scene being associated to nodes in the graph. These nodes are therefore treated as latent [11]. Moreover, a "void" category label is inserted in the dictionary to account for appearance vectors that are not coherent with other categories. When latent patches are involved, BP is run twice, both on the full graph and on a conditioned graph obtained by assigning all the nodes that are labelled in the ground truth. The likelihood to be maximised is

$$\log(L) = \sum_{i=1}^{N} \log(L_i) - \frac{\|\theta\|^2}{2\sigma_\theta^2} \ , \qquad L_i = p(\mathbf{y}_{ai}|\mathbf{X}_i;\theta) \ , \tag{18}$$

where, compared to Eq. (15), we add the term $\frac{\|\theta\|^2}{2\sigma_\theta^2}$ as a weak Gaussian prior imposed to the parameters to control overfitting. The value of the log-likelihood is obtained as

$$\log(L_i) = Z_i(\mathbf{X}_i;\theta) - Z_{ci}(\mathbf{y}_{ai},\mathbf{X}_i;\theta) \ , \tag{19}$$

where the term Z_{ci} is the normalisation factor of the conditioned model for the i-th image, analogous to Eq. (9). The normalisation factors are obtained via sum-product [21], a BP algorithm. This method also gives the marginal probability distributions of single and connected pairs of label variables, that are required to evaluate the value of the likelihood gradient.

Inference is performed via max-sum [21], another BP algorithm that builds the best solution iteratively by locally selecting the best label configurations and propagating the corresponding probability distributions. When run on a tree, both sum-product and max-sum are guaranteed to converge to the exact

result in a time that is proportional to the diameter of the tree. On the contrary, when considering a loopy graph, the generalised version of BP, Loopy BP (LBP), offer no guarantees of convergence and on the quality of the result [22]. The algorithms are reported to converge most of the times to a good approximation of the correct solution in most practical cases. However, since in our model the cost function and its gradient are calculated using a differential form (as in Eq. (19)), errors in one of the graphs introduce inconsistencies in the gradient ascent method that prevent its convergence. For this reason, it is not feasible to use LBP in our method. Another approach would be to optimise an approximation of the likelihood, using methods such as Contrastive Divergence [23].

When weak neighbours are considered, at first the training is performed on the independent patches model, as previously described. Then, the same model is used to label the training set in order to obtain the weak distributions used as features in the training of the CRF model. This approach tends to slightly overestimate the role of the weak neighbours, because the distributions used during training will be generally more accurate than the ones used during testing. However, we have noticed that the independent patches model does not tend to over-fit the training set, especially given the big availability of training examples compared to the small number of parameters to be set.

7 Experiments

As mentioned in the introduction, for the experiments we used the MSRC image dataset. This is a challenging dataset containing mostly outdoor scenes, with the addition of a set of indoor scenes with faces, with cluttered background, multiple object instances and different object scales and degrees of occlusion. The pixel level ground truth labelling is provided. Ambiguous pixels are however left unlabelled ("void"). The database contains 13 semantic categories: "building", "grass", "tree", "cow", "horse", "sheep", "sky", "mountain", "aeroplane", "wa-

Model	Description
IND_{NW}	Independent patches model (Sec. 5.1).
IND	as IND_{NW}, but trained weighting the examples based on the relative category frequencies.
MST_{AC}	CRF model with patches obtained through oversegmentation, connected in a tree obtained via acMST (Sec. 5.2).
$\text{MST}_{\text{AC,B}}$	as MST_{AC}, but with rectangular overlapping patches extracted on a regular grid.
MST_{HUE}	as MST_{AC}, but pairwise potential functions weighted on the difference in feature vectors.
MST_{CC}	as MST_{AC}, but weighting the connections for the MST algorithm in the tree construction phase on the patches centre distance.
MST_{WN}	as MST_{AC}, with the additional contribution of weak neighbours.

Table 1: Description of the different model configurations that have been tested.

	Build. (14.5%)	Grass (30.1%)	Tree (14.1%)	Cow (7.2%)	Sky (13.4%)	Plane (2.8%)	Face (3.2%)	Car (7.5%)	Cycle (7.3%)	Avg.
IND_{NW}	58.4	**93.7**	72.7	54.8	96.1	25.0	54.9	50.2	55.2	72.4
IND	55.4	92.3	74.6	51.8	96.2	34.4	57.5	52.5	59.3	72.6
MST_{AC}	61.0	91.7	81.8	73.4	95.1	72.4	82.8	84.2	85.2	82.8
$\text{MST}_{\text{AC,B}}$	55.2	92.9	84.3	78.8	93.1	75.4	88.9	76.0	75.1	81.0
MST_{HUE}	55.2	92.5	73.8	54.6	95.6	36.9	57.6	53.1	59.5	72.7
MST_{CC}	57.5	91.5	80.3	76.6	94.3	68.1	87.8	77.4	75.0	80.6
MST_{WN}	68.7	93.1	**85.7**	73.5	**96.5**	73.0	**95.8**	85.5	**85.4**	**85.6**
LIT_{gen} [3]	**74.0**	88.7	64.4	77.4	95.7	**92.2**	88.8	81.1	78.7	82.3
LIT_{loc} [11]	71.4	86.8	80.2	**81.0**	94.2	63.8	86.3	**85.7**	77.3	82.3
LIT_{glob} [11]	73.6	91.1	82.1	73.6	95.7	78.3	89.5	84.5	81.4	84.9

Table 2: Models comparison table. Relative category occurrences are shown next to the name, in parenthesis. The configuration associated to each model is detailed thoughout Sec. 7. The results are in terms of percentages of patches correctly classified, for each category (for the reference models, the results are at pixel-level – the difference is negligible given our patch segmentation approach).

ter", "face", "car" and "bicycle". However, we treated the categories "horse", "sheep", "mountain" and "water" as void due to the lack of training data for reliable training and testing phases. Since the database contains only 240 images, to un-bias the results we resorted to 4-fold cross-validation for testing. The dataset has been divided into four subsets containing 25% of the images, and a training has been independently run four times on three subsets, leaving out each time a different subset used for testing. The results have been finally averaged.

To support our claims and to test the validity of the proposals, we performed a set of tests on different configurations of the models. Tests on each config-

uration are used to prove a given point, justifying the related design choice. Therefore, the tested models differ only on a single particular design aspect. The tested models are summarised in Table 1, and the related results are reported in Table 2. Since the MAP training criterion targets the overall rather than single category performance, the latter ones can vary significantly between different configurations. The figures on single category performance are meant to give an idea of the effects of the proposed design choices on the effectiveness in modelling different categories. The precision has been calculated in terms of number of correctly classified patches, where the category of the majority of patch pixels has been considered as the correct one for each patch. Additional comments on the experiments are in the remainder of the section.

The category relative occurrences are very different: the most common one, "grass", occurs in the 30% of the patches, while the rarest amounts to only less than 3% of them. While testing the independent patches model we found that indeed some poorly represented categories suffered from this unbalance, as shown by the first row of Table 2, where $\mathrm{IND_{NW}}$ indicates the model described in Sec. 5.1. We counteracted this effect by introducing a likelihood category weighting vector $\mathbf{w}_c$ whose elements are the reciprocals of the category frequencies in the entire database, or $w_{cj} = 1/p(l_j)$. If the categories distribution in the training image i is represented by the vector $\mathbf{p}_{li}$, the weight for the likelihood of the i-th image is $\mathbf{w}_c \cdot \mathbf{p}_{li}$. Results obtained in this way with the independent patches model are shown in the second row (IND) in Table 2. It is possible to notice a general improvement on the fairness of classification, with a similar overall precision. For this reason, all the other models have been trained by weighting the likelihood as explained.

Then, we ran experiments on structured models. The third row of Table 2, $\mathrm{MST_{AC}}$, uses a CRF as described in Sec. 5.2 with our proposed connection model based on aspect coherence. It is possible to notice a dramatic improvement of the results when compared with the independent patches model. To prove

the validity of our choice for patches, we compared the results obtained with the $\mathrm{MST}_{\mathrm{AC}}$ model with a similar model where however patches are taken in a 20×20 regular grid with 10 pixels overlapping. The results for this model are presented in the row $\mathrm{MST}_{\mathrm{AC,B}}$. It is possible to see how these are globally worse for the block-based model even though the number of patches in this latter model amount to roughly double the number (620 blocks compared to 300 oversegmented patches).

As for the choice of potential functions for the pairwise connections discussed in Sec. 5.2, we tested the functions weighted on the difference on the hue part of the patches' feature vectors. Related results are presented in the row $\mathrm{MST}_{\mathrm{HUE}}$. The reason for the observed under-performance is twofold. The $\mathrm{MST}_{\mathrm{HUE}}$ increases the optimisation problem dimensionality, resulting in a less effective training. Additionally, the pairwise functions are defined on the difference of the appearance vectors between connected variables, that is minimised in the graph construction step. As a result, the pairwise terms are affected to feature vectors noise and their utility is limited.

We also validated the choice of connections: we decided to test a CRF where the graph is built using the MST algorithm weighting the edges on the distances between patch centres ($\mathrm{MST}_{\mathrm{CC}}$). As expected, a drop of performance is observed, especially for those categories that tend to present elongated patches and for which colour is discriminative of the single instances, as "aeroplane", "car" and "bicycle". We then considered full neighbourhoods using the weak neighbours model (indicated as $\mathrm{CRF}_{\mathrm{WN}}$ in Table 2). We had very promising results, showing a clear increase in the overall classification accuracy. Additionally, the increase is spread quite uniformly among all the categories.

Finally, in order to prove the general validity of our method, we compared it with other works that have been tested on the same database [3, 11], and that have been already described in Sec. 2. In particular, we performed better than $\mathrm{LIT}_{\mathrm{gen}}$ [3], a generative approach combining MRF and pLSA. We also

	Void	Build.	Grass	Tree	Cow	Sky	Plane	Face	Car	Cycle
Build.	806	**1336**	23	110	23	36	50	0	143	39
Grass	722	13	**3665**	91	54	1	11	1	3	1
Tree	376	40	99	**1296**	8	17	3	5	7	9
Cow	371	41	28	131	**711**	0	2	11	1	22
Sky	249	50	1	27	5	**1640**	5	0	14	0
Plane	193	119	22	16	3	1	**292**	0	2	3
Face	1193	22	20	2	100	6	0	**378**	0	0
Car	618	159	0	47	18	22	2	0	**790**	20
Cycle	678	79	5	88	0	1	0	15	1	**844**

Table 3: Category confusion matrix for the CRF model with graph based on aspect coherence and weak neighbours. Rows are the system-inferred labels, and columns are the real category labels. The numbers are in terms of patches.

performed better than the CRF-based LIT_{loc} [11]. The limited difference with the LIT_{glob} model [11] is due to the usage, in the latter, of global features that in our configurations were not present. These features are aimed at accounting for large-scale dependencies that in our proposal are not addressed. However, being these features complementary to the ones used in our work, their integration in our model is a viable direction of improvement.

To have a better insight of the performance of our model, in Table 3 we present the category confusion matrix for our best performing model, MST_{WN}. Much of the confusion between categories is due to some of the inter-category aspect similarities being comparable to the intra-category ones. Additionally, in Fig. 5 some examples labelled with the same model are shown. It is possible to notice how the segmentation of objects is generally accurate. The labelling of "void" areas is sometimes reasonable, as for the path in the third example labelled as "building". However, the absence of certain categories causes the overestimation of the extent of some objects, as for the "car" and "bicycle" objects in the sixth and seventh examples, in which the road is absorbed into these objects. Finally, the absence of the "object instance" concept in the system makes the presence of single scattered misclassified patches difficult to tackle, as for the "aeroplane" patch in the second example.

8 Conclusions

In this paper we introduced a method to label image pixels according to their semantic content. This is based on two steps, to separately address short-scale and middle-scale dependencies. For the first ones, a spectral clustering algorithm is used to group adjacent pixels into patches. This helps in finding accurate regions boundaries while providing simplified, aggregated data to the label inference block. This block, that addresses middle-scale dependencies, is based on a CRF, that is, a discriminative probabilistic model, to account for patches context. An important novelty of the method is on the choice of connections between patches in the graphical model in order to perform fast and accurate inference. This choice is based on aspect coherence between neighbouring patches, defined as difference between colour feature vectors. This criterion increases the correlation between linked patches, leading to a sharp performance increase in comparison to other criteria. In connection to the structural choice, we present a second contribution. This is related to the introduction of the concept of *weak neighbours*, that is, patch neighbours not considered directly in the graph to avoid graph loops, but weakly accounted in a two-step classification approach. Experimental results confirm the advantages obtained while considering this additional contextual information. Overall, the presented framework proves to be effective showing good improvements in the labelling process when compared to other works in the literature.

The method presents different interesting directions for improvement, one of the most promising being the utilisation of aspect coherence in association with more complex graphical structures. This can be done directly considering the additional information in the CRF framework, although a modified learning algorithm has to be devised in this case, because an exact inference on the model is no more possible. Another interesting direction for improvement is in the integration of distributed features as in Verbeek *et al*. [11] that have elsewhere proved to significantly improve labelling results.

9 Acknowledgements

The research leading to this paper was supported by the European Commission under contract FP6-027026, Knowledge Space of semantic inference for automatic annotation and retrieval of multimedia content – K-Space, and under the COST Action 292.

References

[1] J. Sivic, B. C. Russell, A. A. Efros, A. Zisserman, and W. T. Freeman, "Discovering objects and their location in images," in *IEEE International Conference on Computer Vision*, vol. 1, pp. 370–377, 2005.

[2] T. Hofmann, "Unsupervised learning by probabilistic latent semantic analysis," *Machine Learning*, vol. 42, no. 1-2, pp. 177–196, 2001.

[3] J. Verbeek and B. Triggs, "Region classification with markov field aspect models," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–8, 2007.

[4] L. Cao and L. Fei-Fei, "Spatially coherent latent topic model for concurrent segmentation and classification of objects and scenes," in *IEEE 11th International Conference on Computer Vision*, pp. 1–8, 2007.

[5] M. Marszałek and C. Schmid, "Spatial weighting for bag-of-features," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2006.

[6] P. Carbonetto, N. Freitas, and K. Barnard, "A statistical model for general contextual object recognition," in *European Conference on Computer Vision*, pp. 350–362, 2004.

[7] I. Ulusoy and C. M. Bishop, "Generative versus discriminative methods for object recognition," in *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 2, pp. 258–265, 2005.

[8] J. Lafferty, A. Mccallum, and F. Pereira, "Conditional random fields: Probabilistic models for segmenting and labeling sequence data," in *18th International Conference on Machine Learning*, pp. 282–289, 2001.

[9] J. Shotton, J. Winn, C. Rother, and A. Criminisi, "Textonboost: Joint appearance, shape and context modeling for multi-class object recognition and segmentation," in *European Conference on Computer Vision*, pp. 1–15, 2006.

[10] X. He, R. S. Zemel, and M. A. Carreira-Perpinan, "Multiscale conditional random fields for image labeling," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2, pp. 695–702, 2004.

[11] J. Verbeek and B. Triggs, "Scene segmentation with crfs learned from partially labeled images," in *Advances in Neural Information Processing Systems*, 2007.

[12] P. Kohli, L. Ladicky, and P. Torr, "Robust higher order potentials for enforcing label consistency," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–8, 2008.

[13] T. Toyoda and O. Hasegawa, "Random field model for integration of local information and global information," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 8, pp. 1483–1489, 2008.

[14] I. Patras, E. A. Hendriks, and R. L. Lagendijk, "Video segmentation by map labeling of watershed segments," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 3, pp. 326–332, 2001.

[15] X. M. He, R. S. Zemel, and D. Ray, "Learning and incorporating top-down cues in image segmentation," in *European Conference in Computer Vision*, May 2006.

[16] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 888–905, 2000.

[17] D. R. Martin, C. C. Fowlkes, and J. Malik, "Learning to detect natural image boundaries using local brightness, color, and texture cues," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 5, pp. 530–549, 2004.

[18] J. Malik, S. Belongie, T. K. Leung, and J. Shi, "Contour and texture analysis for image segmentation," *International Journal of Computer Vision*, vol. 43, no. 1, pp. 7–27, 2001.

[19] J. van de Weijer and C. Schmid, "Coloring local feature extraction," in *European Conference on Computer Vision*, vol. 2, pp. 334–348, 2006.

[20] D. Liu and J. Nocedal, "On the limited memory method for large scale optimization," *Mathematical Programming B*, vol. 45, no. 3, pp. 503–528, 1989.

[21] F. R. Kschischang, B. Frey, and H. Loeliger, "Factor graphs and the sum-product algorithm," *IEEE Transaction on Information Theory*, vol. 47, no. 2, pp. 498–519, 2001.

[22] B. J. Frey and D. J. Mackay, "A revolution: Belief propagation in graphs with cycles," in *Advances in Neural Information Processing Systems*, vol. 10, 1997.

[23] G. E. Hinton, "Training products of experts by minimizing contrastive divergence," *Neural Computation*, vol. 14, no. 8, pp. 1771–1800, 2002.

Figures

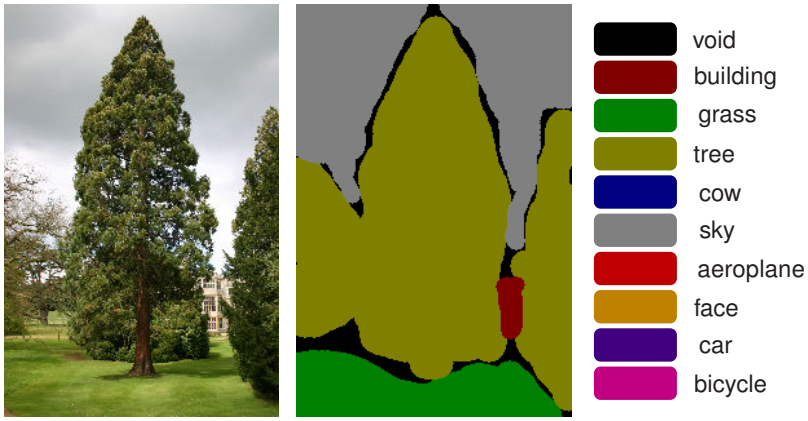

Figure 1: Semantic segmentation with, from the left, input image, segmented image ground truth, and label colours legend for all the categories present in the MSRC dataset used in this paper.

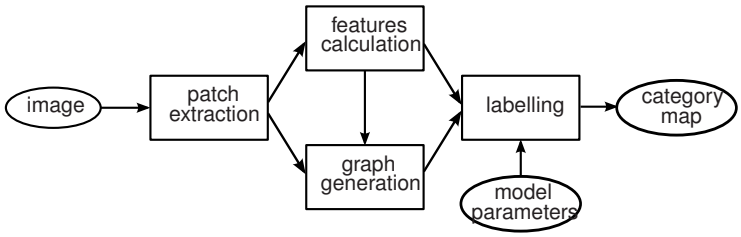

Figure 2: Block representation of the proposed system.

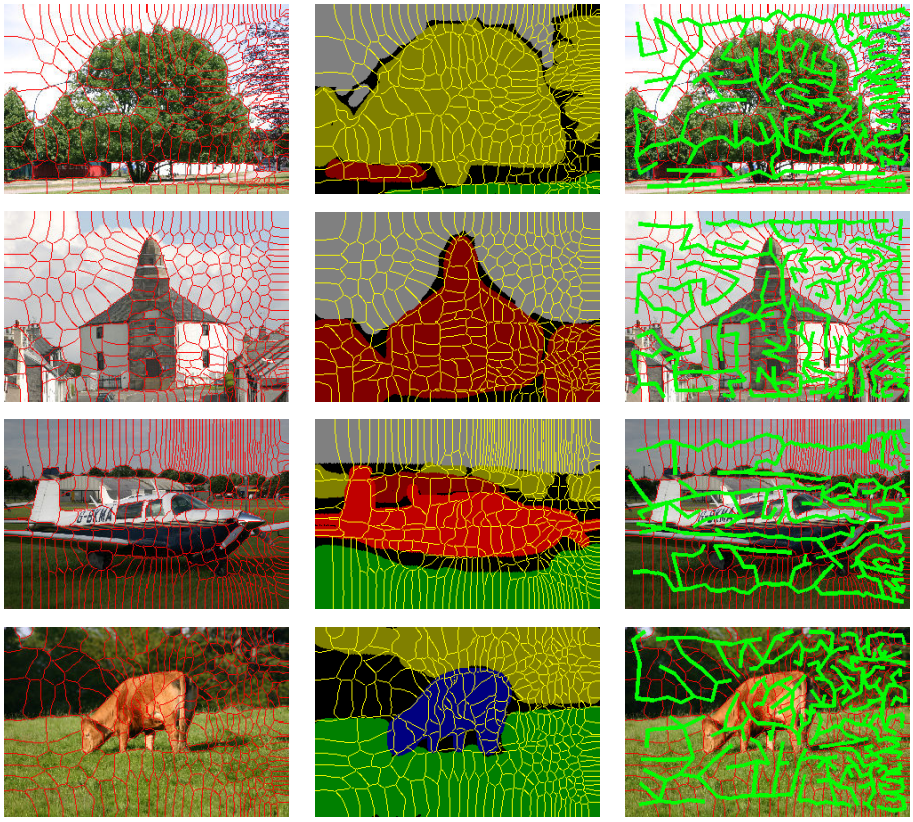

Figure 3: Oversegmentation (300 patches) using NCuts (original images size 321×214). In the central column, the ground-truth for the corresponding images is displayed, with the superimposed segmentation. Finally, in the right column is the Minimum Spanning Tree based on aspect coherence built over segmented images.

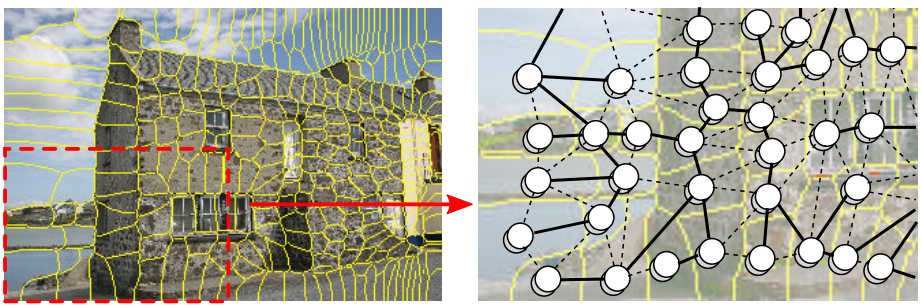

Figure 4: Example of the used tree structure. The white nodes are the nodes of the CRF. The solid connections are the aspect-coherence-based MST. The dashed connections are weak neighbourhoods.

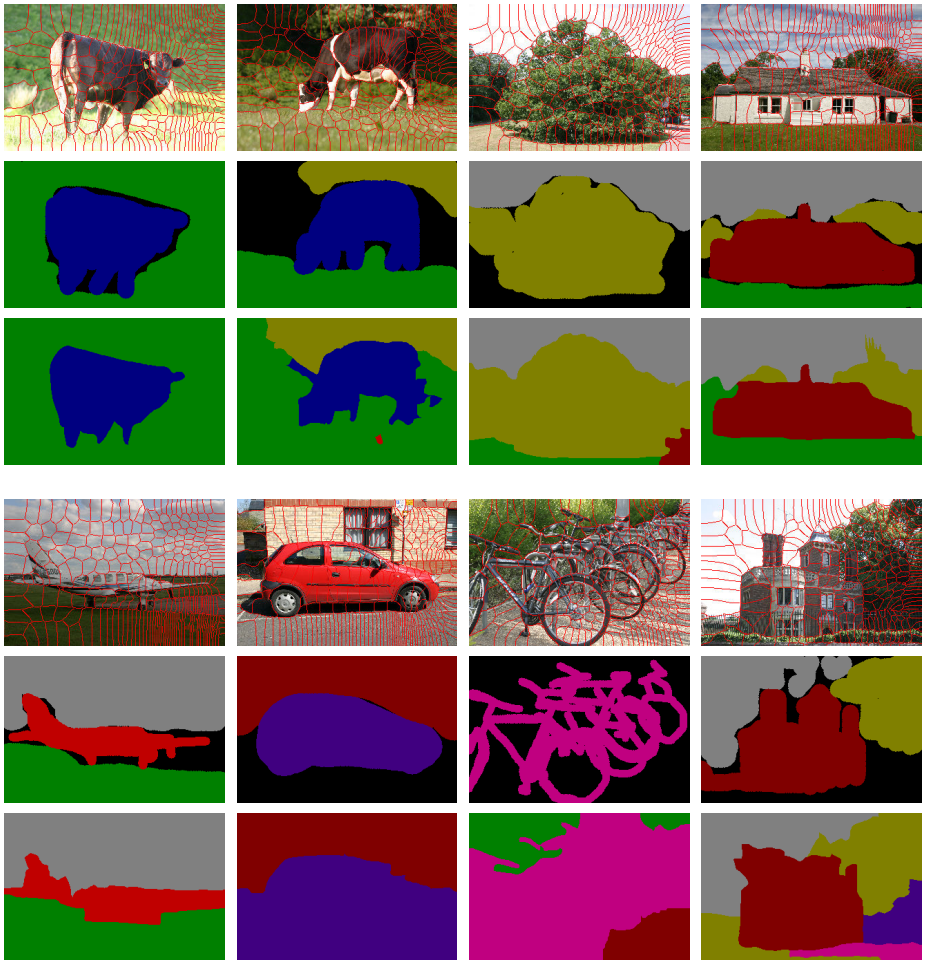

Figure 5: Images from the MSRC database segmented with the proposed method. In the first row the original images, on the second row the ground truth, on the third row our results with the CRF_{WN} model. The category labels legend is reported in Fig. 1