Semantic Image Labelling as a Label Puzzle Game

Peter Kontschieder[1]
kontschieder@icg.tugraz.at
Samuel Rota Bulò[2]
srotabul@dsi.unive.it
Michael Donoser[1]
donoser@icg.tugraz.at
Marcello Pelillo[2]
pelillo@dsi.unive.it
Horst Bischof[1]
bischof@icg.tugraz.at

[1] Institute for Computer Graphics and Vision
Graz University of Technology
Austria

[2] Dipartimento di Scienze Ambientali, Informatica e Statistica
Università Ca' Foscari Venezia
Italy

Abstract

In this work we introduce a novel solution to the semantic image labelling problem, *i.e*. the task of assigning semantic object class labels to individual pixels in a test image. Conventional methods are typically relying on random fields for modelling interactions between neighboring pixels and obtaining smooth labelling results using unary and pairwise cost functions. Instead, we consider the labelling problem as a puzzle game, where the final labelling is obtained by assembling discriminatively learned candidate sets of label puzzle pieces, each representing a topological and semantically plausible label configuration. The puzzle game is set up by means of a modified random forest classifier, designed to learn the local, topological label-structure and hence the local context associated to the training data. To solve the puzzle game we propose an iterative optimization technique that maximizes an agreement function by alternatingly seeking for the best label puzzle piece per pixel and the resulting semantic labelling per image. We provide both, theoretical properties of our puzzle solver algorithm as well as experimental results on the challenging MSRC and CamVid databases. In a direct comparison with a conditional random field we obtain superior results, indicating the practicability of our proposed method.

1 Introduction

The field of semantic image labelling has received great attention and evolved in a remarkable manner during the past couple of years. Given an input image, it aims for the proper assignment of a-priori learned class labels to each pixel in a test image[1]. For instance, a typical street scene might result in coherently labelled regions of road, car, bicyclist and so on. In order to obtain labellings reflecting the natural statistics of a scene, state-of-the-art approaches [13, 18, 19] combine multiple, complementary cues at different levels within

[1] For brevity reasons we use the terms *image labelling* or solely *labelling* interchangeably instead of *semantic image labelling* throughout the paper.

BMVC 2011 http://dx.doi.org/10.5244/C.25.11.1

random field models [20]. These include low-level cues which are mostly computed on a per-pixel basis and incorporate local color or texture statistics. Mid-level cues operate on regions or superpixels to provide shape, continuity or symmetry information. Motivated from perceptual psychology [2, 3], high-level cues introduce global image statistics and information about inter-object or contextual relations, seeking for proper scene configurations at the image level.

Many researchers are following the idea of Shotton *et al*. [31], putting special emphasis on the generation of good unary potentials to be used in conjunction with a conditional random field (CRF) model. Indeed, unary potentials have a significant influence on the success of a labelling algorithm. For instance, [31] uses an adapted version of joint boosting [35] to train unary classifiers integrating textons [23] and shape filters. Similarly, [33] uses a boosting approach to combine an extended set of different feature cues. Recently, several methods [7, 15, 29, 32, 36] used random forests [1, 6, 12] for obtaining properly learned combinations of local features like color, intensity derivatives, covariance features [26], textons, HOG features [10] or motion and 3D structure features [7].

To render the inference process more efficient and include segmentation information, many works [21, 28] consider *mid-level* or super-pixel representations rather than individual pixels. In such a way, the labelling problem is defined over regions, mostly obtained by Mean Shift [9], graph-based approaches [11] or Normalized Cut [30]. To cope with suboptimal segmentations not following the desired object boundaries, [17, 25] combine multiple segmentations or reshape superpixels to recover from errors as presented in [14]. Recently, some researchers [18, 19, 27, 34] also started to incorporate *high-level* information, *e.g*. contextual (semantic) or object detector information, into CRFs. This yields a considerable improvement of the overall labelling results, since contradicting labellings can be resolved by means of global or co-occurrence statistics. In other words, such information helps to improve the labelling of adjacent regions being partially labelled by non-compatible object class configurations.

The label space characterizing an image labelling problem instance does indeed exhibit an inherently topological structure which renders the class labels explicitly interdependent. Many approaches to labelling, however, are not exploiting this information properly, as they rely on classifiers trained on a set of labeled images, which associate pixels only with single, *atomic* class labels acting as arbitrary identifiers without any dependencies among them (*e.g*. [7, 15]). In this way, the structured label space information in the training images remains largely unexploited. As a consequence, labellings obtained at a low-level are quite noisy and exhibit configurations of labels which never appeared in the training images. To alleviate this effect, the atomic labels obtained by the base classifiers are combined in a more or less sophisticated way, *e.g*. by means of a CRF. However, the rules guiding this label relaxation process are typically imposed in a top-down fashion rather than being learnt from training data. At a high-level, some efforts have been put on capturing topological relationships between labels, *e.g*. by collecting co-occurrence statistics of categories in images. However, the integration of this information in the labelling approaches leads typically to simplistic, but expensive, high-order energy terms in CRFs, or to a-posteriori re-elaboration of low- or mid-level labelling solutions.

Contributions. In this paper we propose a novel method to include local, contextual information into the low-level classification process. Instead of integrating a series of complementary cues within a random field model, we formulate the image labelling problem as the task of assembling topological label information in a coherent way. Intuitively, our approach

can be explained as a puzzle game where the puzzle pieces are represented by structured object class labels. These structured labels are directly obtained and learned from the ground truth training data and always exhibit a semantically meaningful label configuration. In such a way, the set of possible label puzzle pieces only shows plausible label configurations such as a cow standing on grass but not on water. In concrete terms, the labelling of an image is obtained as a result of a two-stage process. First, the label puzzle game is set up by assigning a set of plausible puzzle pieces to each pixel in the test image, using a modified random forest classifier. Afterwards, we search for a solution of the label puzzle, which consists of both, a per-pixel class label and puzzle piece selection, maximizing a measure of overall agreement. This optimization problem is addressed by means of a heuristic, which alternates between optimizing the image labelling and the per-pixel puzzle piece selection. As a result, we obtain a joint labelling based on the selection of plausible, local label configurations, respecting the local contextual information of neighboring pixels.

Paper organization. In Section 2 we describe our image labelling puzzle approach, and provide related definitions and notations. In Section 3 we describe how to set up a label puzzle game by means of a modified random forest classifier, which discriminatively learns structured labels from the training data. In Section 4 we introduce an algorithm to solve the label puzzle game, show its theoretical properties and analyse its complexity. Finally, we provide experimental results and concluding remarks in Sections 5 and 6, respectively.

2 The Label Puzzle Game

In this section we propose our novel idea, which considers image labelling as the task of assembling a kind of puzzle, where the pieces are label configurations (*e.g.* in our experiments they are square patches of labels) gathered from the training images during a learning and classification process. Please note that this is in contrast to a common tiling or jigsaw puzzle [8, 37], where the pieces form a partition of the target image in the image domain. Instead, we associate each pixel with a possibly different set of label puzzle pieces in the label domain from which only one must be selected. Additionally, pieces belonging to different pixels may overlap. Given a test image, the goal is to simultaneously assemble the related puzzle and assign labels to pixels in a way such that the agreement of the selected pieces with the underlying labelling is maximized.

Notations and definitions. An *image* is a function $f : D \to \mathbb{R}^d$ mapping pixels in $D \subseteq \mathbb{Z}^2$ to d-dimensional feature vectors, encoding different local cues of the image (*e.g.* color, gradient features, filter banks). A *labelling* for an image is a function $\ell : D \to Y$ mapping pixels to labels in $Y = \{1, \ldots, k\}$. A (label) puzzle piece is a (local) *label configuration*, *i.e.* a function $p : \mathbb{Z}^2 \to Y \cup \{\perp\}$ mapping two-dimensional points to labels or to void ($\perp$), a special symbol indicating the absence of a label. The set of images, labellings and puzzle pieces (*i.e.* label configurations) are denoted by $\mathscr{I}$, $\mathscr{L}$ and $\mathscr{P}$, respectively. A *puzzle configuration* is a function $z : D \to \mathscr{P}$ associating each pixel in D with a puzzle piece in $\mathscr{P}$. The set of puzzle configurations is denoted by $\mathscr{Z}$. Note that, for notational convenience, we will write in the sequel $z_{i,j} \in \mathscr{P}$ instead of $z(i, j)$ and we will denote with $z_{i,j}(u, v)$ the label in position (u, v) in puzzle piece $z_{i,j}$.

The *agreement* of a puzzle piece $p \in \mathscr{P}$ located in $(i, j) \in D$ with a labelling $\ell \in \mathscr{L}$ is defined as the number of corresponding pixels sharing the same label, *i.e.*

$$\phi^{(i,j)}(p, \ell) = \sum_{(u,v) \in D} [p(u - i, v - j) = \ell(u, v)] \,, \tag{1}$$

where $[P]$ are the Iverson brackets yielding 1 if proposition P is true, 0 otherwise. Given a puzzle configuration $z \in \mathscr{Z}$ and a labelling $\ell \in \mathscr{L}$, the *total agreement* $\Phi(z,\ell)$ of the image labelling puzzle is the sum of the agreements of each puzzle piece in z with the labelling ℓ, *i.e.*

$$\Phi(z,\ell) = \sum_{(i,j)\in D} \phi^{(i,j)}(z_{i,j},\ell) \,. \tag{2}$$

The label puzzle game. A *label puzzle game* for an image $f \in \mathscr{I}$ is a function π_f mapping each pixel $(i,j) \in D$ to a non-empty set of puzzle pieces $\pi_f(i,j) \subseteq \mathscr{P}$. This function restricts the possible choices of puzzle pieces per pixel and, hence, also the set of admissible puzzle configurations to

$$\mathscr{Z}|_{\pi_f} = \left\{ z \in \mathscr{Z} \,\middle|\, z_{i,j} \in \pi_f(i,j) \right\} .$$

A solution of a label puzzle game π_f is a pair $(z^*,\ell^*) \in \mathscr{Z}|_{\pi_f} \times \mathscr{L}$ consisting of an admissible puzzle configuration and a labelling for f yielding the maximum total agreement:

$$(z^*,\ell^*) \in \arg\max_{(z,\ell)} \left\{ \Phi(z,\ell) \,\middle|\, (z,\ell) \in \mathscr{Z}|_{\pi_f} \times \mathscr{L} \right\} . \tag{3}$$

A heuristic for finding a solution of (3) will be discussed in Section 4.

An important component of our framework is the *label puzzle game generator* providing the label puzzle game π_f for any image $f \in \mathscr{I}$ we want to label. The generator is obtained as the result of a supervised learning process, involving a set of labelled training images, and will be discussed in the next section.

3 Label Puzzle Game Generator

In this section, we describe a puzzle game generator built upon an adapted random forest classifier [16]. The basic idea is to collect a set of admissible label puzzle pieces for each pixel in a test image, which will be used to create the label puzzle game. To this end, we augment random forests with the ability of performing structured label predictions rather than single and atomic classifications. In such a way we directly obtain structured labels as output of our classifier, subsequently denoted as *puzzle pieces*.

Before moving into the details of our approach, we briefly review the traditional random forest framework [1, 17]. A random forest is an ensemble of binary decision trees, each of which is a classifier mapping samples in $\mathscr{X}$ to class labels in Y. In the context of image labelling, the sample space $\mathscr{X}$ consists of a set of labelled patches (*e.g.* square regions of pixels) extracted from the training images, where each patch is associated with the label of a specific pixel it contains (typically the one in the center). The prediction for a sample in a decision tree takes place by routing it from the root node to a leaf holding a class label. The path followed by a sample moving along the tree is determined by split functions $\psi : \mathscr{X} \to \{\text{left}, \text{right}\}$ located in each node, according to which a sample is forwarded to the left or right child. The prediction for the whole forest is computed using a majority vote criterion from the predictions cast by its single decision trees. As for the learning part, decision trees in a random forest are recursively trained by selecting in each node a split function from a set of randomly generated ones, which induces a partition of the training set showing the best information gain about the class label distributions due to the split. According to the chosen split function in a node, the training set is then partially forwarded to its left and right child, respectively. If the training samples reaching a node are less than a given threshold, if they exhibit a low entropy in their class label distribution or if a maximum

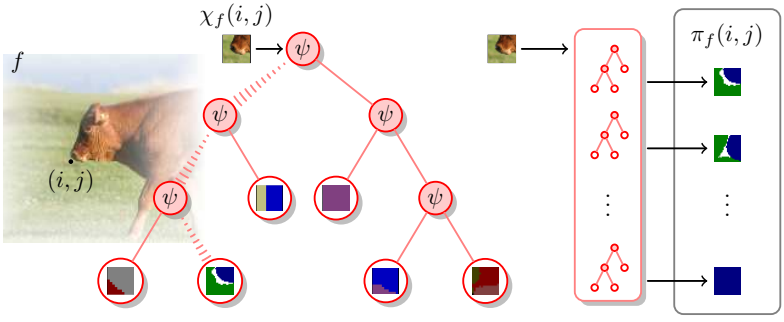

Figure 1: Pipeline of the construction of a label puzzle game. For each pixel (i,j) of the test image f, we extract an image patch $\chi_f(i,j)$ and compute a set of plausible puzzle pieces $\pi_f(i,j)$ for it by means of a modified random forest.

depth in the tree is reached, the recursion stops and a leaf is grown. Finally, the class label best represented in the training samples is assigned to the leaf.

To adapt the random forest to collect puzzle pieces, we change the label space from Y to the set of puzzle pieces $\mathscr{P}$. Hence, each decision tree can be considered as a function $h : \mathscr{X} \to \mathscr{P}$ mapping image patches in $\mathscr{X}$ to puzzle pieces in $\mathscr{P}$. Accordingly, the training set consists of image patches with a corresponding structured label, *i.e.* a puzzle piece $p \in \mathscr{P}$ collected from the ground truth. The shape of a puzzle piece may be arbitrary but, for simplicity, we assume all puzzle pieces to have the same shape. Note that in our experiments we considered simple square regions of labels as puzzle pieces, as illustrated in Figure 1.

Besides dealing with a structured label space, our decision trees present other significant differences with respect to standard ones. First, we changed the way the best split function is selected at each tree node in order to take the new label space into account. Specifically, we randomly select for each node a point $(i,j) \in \mathbb{Z}^2$ and any training sample $(x,p) \in \mathscr{X} \times \mathscr{P}$ reaching that node is given label $p(i,j)$. By so doing, the same training sample may be considered with different labels in different nodes, thereby exploiting the whole structure of the puzzle piece during the tree construction. Moreover, the split function selection can still be efficiently carried out using, *e.g.* the technique based on information gain. A second difference is in the way a puzzle piece is selected as representative in a leaf: Given a leaf of the tree, let $T \subseteq \mathscr{X} \times \mathscr{P}$ be the subset of the training set that reached the leaf during the training procedure. Since we would like to select a representative close to the mode of the distribution of puzzle pieces in the leaf, we estimate a conditional probability $\Pr(p|T)$ of a puzzle piece given T. For simplicity, we make a pixel independence assumption, thereby obtaining:

$$\Pr(p|T) = \prod_{(i,j)\in\mathbb{Z}^2} \Pr^{(i,j)}(p(i,j)|T) \,,$$

as product of the marginal class label distributions $\Pr^{(i,j)}(y|T)$ of pixels in position (i,j) given T, where

$$\Pr^{(i,j)}(y|T) = \frac{1}{|T|} \sum_{(x,p)\in T} [p(i,j) = y] \,.$$

The puzzle piece representative p^* for the leaf is then selected as the one maximizing the joint probability over the set of available puzzle pieces:

$$p^* \in \arg\max_p \{\Pr(p|T) \,|\, (x,p) \in T \text{ for some } x \in \mathscr{X}\} \,.$$

Finally, a random forest $\{h_1,\ldots,h_n\}$ consisting of n trees can be considered as a function H mapping image patches $x \in \mathscr{X}$ to non-empty sets of puzzle pieces $H(x) \subseteq \mathscr{P}$ in the following way:

$$H(x) = \bigcup_{k=1}^{n} \{h_k(x)\} .$$

Note that, as opposed to standard random forests, the predictions gathered from the single decision trees are not merged into a single puzzle piece, but we keep them all as a set of puzzle pieces. The modified random forest can then be used to generate a label puzzle game for an image $f \in \mathscr{I}$ as follows:

$$\pi_f(i,j) = H(\chi_f(i,j)) , \tag{4}$$

where $\chi_f(i,j) \in \mathscr{X}$ denotes the patch extracted from image $f \in \mathscr{I}$ in position $(i,j) \in D$. In Figure 1, we summarize the label puzzle game generation process for a particular image.

4 Label Puzzle Game Solver

The optimization problem in (3) underlying our image labelling approach is in general non-trivial to solve. The algorithm we propose in this section is a heuristic, which is simple and effective as shown in the experiments conducted (see Section 5). It is based on an alternating optimization technique, where we iteratively switch between optimizing the labelling variable $\ell \in \mathscr{L}$ and the puzzle configuration variable $z \in \mathscr{Z}|_{\pi_f}$.

Let $\ell^{(t)}$ be the labelling of the image at a given time $t \geq 0$. The puzzle configuration $z^{(t+1)}$ at time $t+1$ can be obtained according to the following updating scheme:

$$z_{i,j}^{(t+1)} \in \arg\max_{p} \left\{ \phi^{(i,j)}\left(p, \ell^{(t)}\right) \middle| \, p \in \pi_f(i,j) \right\} , \tag{5}$$

which selects for each pixel $(i,j) \in D$ a puzzle piece in the set $\pi_f(i,j)$ maximizing the agreement with the labelling $\ell^{(t)}$. On the other hand, given the puzzle configuration $z^{(t+1)} \in \mathscr{Z}|_{\pi_f}$ at time $t+1$, we compute the new labelling $\ell^{(t+1)} \in \mathscr{L}$ by taking a majority vote over all puzzle pieces as follows:

$$\ell^{(t+1)}(u,v) \in \arg\max_{y} \left\{ \sum_{(i,j)\in D} \left[z_{i,j}^{(t+1)}(u-i, v-j) = y \right] \middle| \, y \in Y \right\} . \tag{6}$$

The iterative process is started from an initial labelling $\ell^{(0)} \in \mathscr{L}$ and, by repeatedly applying rules (5) and (6), it will eventually converge towards a local solution of (3). Theorem 1, indeed, provides a theoretical guarantee that the iterative scheme never decreases the value of the objective function Φ.

Theorem 1. *Let π_f be a label puzzle game for image $f \in \mathscr{I}$, let $\ell^{(0)} \in \mathscr{L}$ be an initial labelling for f, and let $z^{(0)} \in \mathscr{Z}|_{\pi_f}$ be an initial puzzle configuration. Then for any $t \geq 0$ we have*

$$\Phi\left(z^{(t+1)}, \ell^{(t)}\right) \geq \Phi\left(z^{(t)}, \ell^{(t)}\right) \tag{7}$$

and

$$\Phi\left(z^{(t+1)}, \ell^{(t+1)}\right) \geq \Phi\left(z^{(t+1)}, \ell^{(t)}\right) \tag{8}$$

where $z^{(t+1)}$ and $\ell^{(t+1)}$ are computed according to (5) and (6), respectively.

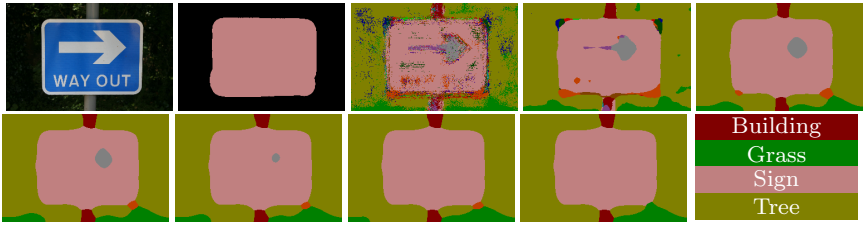

Figure 2: A labelling example of the proposed approach. Top to bottom, left to right: Image to be labelled, groundtruth labelling, initial random forest classification, labellings obtained by our approach after $t = 0, 5, 10, 20, 35, 50$ iterations, final label captions.

Proof. By (5) we have for all $t \geq 0$ and $(i, j) \in D$:

$$\phi\left(z_{i,j}^{(t+1)}, \ell^{(t)}\right) \geq \phi\left(z_{i,j}^{(t)}, \ell^{(t)}\right).$$

By summing up each side of this inequality for all pixels $(i, j) \in D$ we obtain (7).

As for the second inequality, note that by (6) and (1) we have

$$\sum_{(i,j)\in D}\left[z_{i,j}^{(t+1)}(u-i, v-j) = \ell^{(t+1)}(u,v)\right] \geq \sum_{(i,j)\in D}\left[z_{i,j}^{(t+1)}(u-i, v-j) = \ell^{(t)}(u,v)\right]$$

for all $(u, v) \in D$. This together with a trivial re-ordering of the summations yields

$$\begin{aligned}\Phi\left(z^{(t+1)}, \ell^{(t+1)}\right) &= \sum_{(u,v)\in D}\sum_{(i,j)\in D}\left[z_{i,j}^{(t+1)}(u-i, v-j) = \ell^{(t+1)}(u,v)\right] \\ &\geq \sum_{(u,v)\in D}\sum_{(i,j)\in D}\left[z_{i,j}^{(t+1)}(u-i, v-j) = \ell^{(t)}(u,v)\right] = \Phi\left(z^{(t+1)}, \ell^{(t)}\right)\end{aligned}$$

from which the result derives. □

As for the computational complexity of the solver, let N be the number of pixels, K the average number of puzzle pieces per pixel, M the number of non-void elements of a puzzle piece, k the number of labels, and γ the number of iterations. An update step for the pixel configuration z has complexity $O(K \cdot M \cdot N)$, while an update step for the labelling ℓ has complexity $O((k+M) \cdot N)$. The overall complexity is thus given by $O(\gamma \cdot (k+M+KM) \cdot N)$. Note that in our experiments we stopped the iterative process if either a fixed point or a maximum number $\gamma = 75$ of iterations was reached.

5 Experiments

In order to demonstrate the quality of our method, we evaluate on the challenging and widely known CamVid [7] and MSRCv2 [31] databases. We use almost the same setup for both databases, *i.e.* we collect the training samples on a regular grid with a stride of 10 (CamVid) or 5 (MSRCv2) and apply an inverse weighting scheme to correct the imbalance of the training sample distribution. We train forests consisting of 15 decision trees with 500 iterations per node test, stopping when less than 5 samples were available per leaf node. The feature patch sizes are fixed to 20×20 while we evaluate different puzzle piece sizes on the MSRCv2 database.

We use the following feature cues: CIELab raw channel intensities, first and second order derivatives of the luminance channel and correlation coefficients between covariances

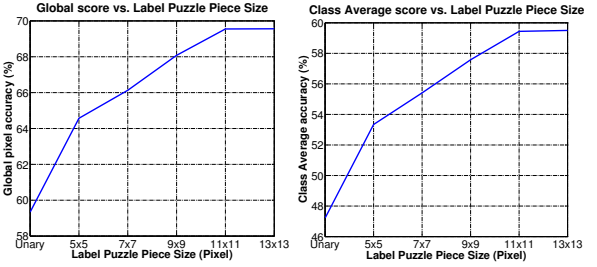

Figure 3: Evaluation of label puzzle piece sizes and their impact on the global pixel (left) and class average (right) accuracies for the MSRCv2 database.

of the RGB raw channel intensities and the first order derivatives of the grayscale intensity image, similar to [15, 26]. The pixel-wise classifications (*unary*) are computed according to the class label distribution of the central pixels of the puzzle piece returned by the trees. In order to obtain the initial pixelwise labelling $\ell^{(0)}$ for the puzzle solver, we take a majority vote decision based on the label statistics collected over all overlapping puzzle pieces. For performance evaluation, we use standard criteria as *e.g.* used in [31]. These are the *Global Pixel Average*, *i.e.* the fraction of correctly classified pixels computed over all classes and test images, and the more strict *Class Average*, defined as the fraction of correctly classified pixels belonging to a specific category over all test images.

On both databases, we compare to the labelling results obtained by minimizing the energy term of a conditional random field (CRF) model with graph cuts, when supplied with our random forest classification results. We use the publicly available GCO implementation[2] [5] and the alpha expansion solver. As unary or data terms, we provide the central label statistics over the puzzle pieces of the entire forest. For the pairwise or smoothness term we use the standard, contrast-sensitive Potts model as suggested in [4].

5.1 MSRCv2 Database

This database consists of 532 images containing 21 object classes and predefined splits into 276 training and 256 test images as suggested in [31]. Our random forests obtain pixel classification scores of $(59.3/47.2\%)$ (global/class average) which are higher than the scores obtained by related random forest approaches of Kluckner *et al.* [15] $(55.8/42.2\%)$, the naive, supervised approach of Shotton *et al.* [32] $(49.7/34.5\%)$ and Lazebnik *et al.* [22] $(53.3/40.7\%)$ using combinations of color, textons and SIFT [24] features. There are however methods starting with a significantly better baseline as in Schroff *et al.* [29] $(69.7/-\%)$ which we were not able to reproduce.

Influence of puzzle piece sizes In Figure 3 we show the influence of the puzzle piece size on the obtained classification scores. The correlation between label puzzle size and classification score is clearly indicated for both performance measures. This strengthens our initial assumption that the introduction of contextual information at the local level is viable for image labelling. Further increase of the label puzzle pieces will likely introduce smoothing effects along the object boundaries unless a sufficient amount of label transitions are captured during the training phase.

Comparison to CRF As illustrated in Table 5.1, we obtain superior results for both, the global pixel labelling accuracy and the more strict per-class average score when compared to a CRF. With our method we always improve over the baseline classification and are superior

[2] http://vision.csd.uwo.ca/code/

Method	Global	Class Avg	Building	Grass	Tree	Cow	Sheep	Sky	Aeroplane	Water	Face	Car	Bicycle	Flower	Sign	Bird	Book	Chair	Road	Cat	Dog	Body	Boat
Unary	59	47	28	93	75	53	62	94	44	53	63	36	57	60	32	19	44	19	61	36	29	25	5
Unary + CRF	67	57	28	96	83	66	74	93	58	56	70	45	**81**	**80**	39	23	**64**	**32**	75	**56**	**42**	31	4
Unary + Puzzle	**70**	**60**	**43**	96	83	**78**	**81**	**96**	**70**	**59**	**71**	**55**	79	73	**42**	**25**	59	29	75	53	40	**37**	**6**

Table 1: Comparison of scores on MSRCv2 database in [%] for puzzle piece size of 11×11. *Unary* are the scores obtained by the structured random forest classifications alone, *Unary + CRF* are the results using the conditional random field and *Unary + Puzzle* refers to the final labelling result, obtained by our proposed method. Bold style indicates best score while underlined scores are same among CRF and our approach.

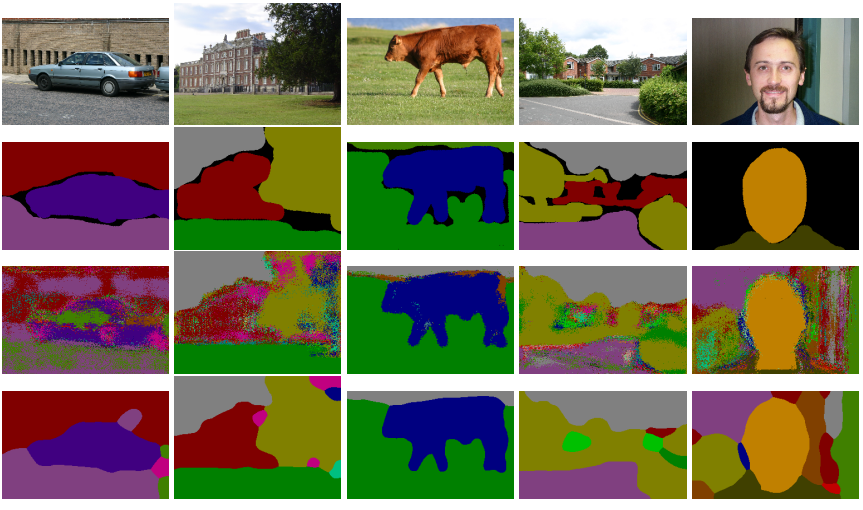

Figure 4: Qualitative labelling results obtained by our method on the MSRCv2 database. First row: Original image, Second row: Ground truth labelling, Third row: Unary classifications, Last row: Our proposed labelling approach.

or equal to the CRF in 15/21 classes. Our final labelling result is in a comparable range of those reported in [15, 22, 29, 31]. We are aware that state-of-the-art methods [18] achieve higher scores on the MSRCv2 database, however, they are using higher-order terms in the CRF and globally learned contextual information while we deliberatly restrict our method to local classification results.

5.2 CamVid Database

The Cambridge-driving Labeled Video Database (CamVid) [7] is a collection of videos captured on road driving scenes, consisting of more than 10 minutes of high quality (970×720), 30 Hz footage. A subset of 711 images is almost entirely annotated into 32 categories, however, in our experiments on this database we used only the 11 commonly used categories with the same splits for training and testing as presented in [7, 33]. Our unary classifaction results are (70.7/44.8%) (global/class average) which are improved to (75.0/47.8%) when using the CRF model. However, with our proposed label puzzle approach we can boost the scores to (81.7/49.6%), showing competitive results in comparison to Brostow *et al.* [7] (69.1/53.0%), and Sturgess *et al.* [33] (76.4/59.8%) and (79.8/59.9%) in a CRF setting with only unary terms and unary+pairwise terms, respectively.

6 Conclusion

In this paper we have proposed a novel approach for the task of image labelling, which allows to exploit local contextual information and the label topological structure observed in the training data. This is achieved by defining a label puzzle game, where a final labelling is obtained by maximizing the mutual agreement of structured class labels (our label puzzle pieces), which are associated with every pixel. We introduced a modification of the random forest classifiers in order to discriminatively learn and provide the structured class labels needed for the construction of a label puzzle game. We showed how the optimization problem underlying our approach can be optimized in order to obtain the final labelling, and we provided theoretical properties and a complexity analysis of our algorithm. Our approach achieved superior results in experiments on the MSRCv2 and CamVid databases when directly compared to a standard CRF formulation, supporting our claim that high-quality labelling results can be obtained by properly learning and integrating local contextual information at a low-level. As a future work, we plan to extend our approach by incorporating additional mid-level cues and global co-occurrence statistics.

Acknowledgements We acknowledge the financial support of the Austrian Science Fund (FWF) from project Fibermorph (P22261-N22), the Research Studios Austria Project μSTRUCSCOP (818651) and the Future and Emerging Technology (FET) Programme within the Seventh Framework Programme for Research of the European Commission, under FET-Open project 'SIMBAD' (213250).

References

[1] Y. Amit and D. Geman. Shape quantization and recognition with randomized trees. *Neural Computation*, 9(7):1545–1588, 1997.

[2] I. Biederman. Perceiving real-world scenes. *Science*, 177(43):77–80, 1972.

[3] I. Biederman, R. J. Mezzanotte, and J. C. Rabinowitz. Scene perception: detecting and judging objects undergoing relational violations. *Cognitive Psychology*, 14(2), 1982.

[4] Y. Boykov and M. P. Jolly. Interactive graph cuts for optimal boundary and region segmentation of objects in n-d images. *(ICCV)*, 2001.

[5] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *(PAMI)*, 2001.

[6] L. Breiman. Random forests. In *Machine Learning*, pages 5–32, 2001.

[7] G. J. Brostow, J. Shotton, J. Fauqueur, and R. Cipolla. Segmentation and recognition using structure from motion point clouds. In *(ECCV)*, 2008.

[8] T. S. Cho, S. Avidan, and W. T. Freeman. A probabilistic image jigsaw puzzle solver. In *(CVPR)*, 2010.

[9] D. Comaniciu and P. Meer. Mean shift: A robust approach toward feature space analysis. *(PAMI)*, 2002.

[10] N. Dalal and B. Triggs. Histogram of oriented gradients for human detection. In *(CVPR)*, 2005.

[11] P. F. Felzenszwalb and D. P. Huttenlocher. Efficient graph-based image segmentation. *(IJCV)*, 2004.

[12] P. Geurts, D. Ernst, and L. Wehenkel. Extremely randomized trees. *Machine Learning*, 36(1):3–42, 2006.

[13] J. M. Gonfaus, X. Boix, J. van de Weijer, A. D. Bagdanov, J. Serrat, and J. Gonzalez. Harmony potentials for joint classification and segmentation. In *(CVPR)*, 2010.

[14] S. Gould, R. Fulton, and D. Koller. Decomposing a scene into geometric and semantically consistent regions. In *(ICCV)*, 2009.

[15] S. Kluckner, T. Mauthner, P. M. Roth, and H. Bischof. Semantic image classification using consistent regions and individual context. In *(BMVC)*, 2009.

[16] P. Kontschieder, S. Rota Bulò, H. Bischof, and M. Pelillo. Structured class-labels in random forests for semantic image labelling. In *(ICCV)*, 2011.

[17] L. Ladicky, C. Russell, P. Kohli, and P. H. S. Torr. Associative hierarchical CRFs for object class image segmentation. In *(ICCV)*, 2009.

[18] L. Ladicky, C. Russell, P. Kohli, and P. H. S. Torr. Graph cut based inference with co-occurrence statistics. In *(ECCV)*, 2010.

[19] L. Ladicky, P. Sturgess, K. Alahari, C. Russell, and P. H. S. Torr. What, where & how many? Combining object detectors and CRFs. In *(ECCV)*, 2010.

[20] J. Lafferty, A. Mccallum, and F. Pereira. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *(ICML)*, 2001.

[21] D. Larlus and F. Jurie. Combining appearance models and markov random fields for category level object segmentation. In *(CVPR)*, 2008.

[22] S. Lazebnik and M. Raginsky. An empirical bayes approach to contextual region classification. In *(CVPR)*, 2009.

[23] T. Leung and J. Malik. Representing and recognizing the visual appearance of materials using three-dimensional textons. *(IJCV)*, 2001.

[24] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *(IJCV)*, 2004.

[25] T. Malisiewicz and A. A. Efros. Improving spatial support for objects via multiple segmentations. In *(BMVC)*, 2007.

[26] F. Porikli, O. Tuzel, and P. Meer. Covariance tracking using model update based on lie algebra. In *(CVPR)*, 2006.

[27] A. Rabinovich, A. Vedfaldi, C. Galleguillos, E. Wiewiora, and S. Belongie. Objects in context. In *(ICCV)*, 2007.

[28] B. Russell, W. Freeman, A. Efros, J. Sivic, and A. Zisserman. Using multiple segmentations to discover objects and their extent in image collections. In *(CVPR)*, 2006.

[29] F. Schroff, A. Criminisi, and A Zisserman. Object class segmentation using random forests. In *(BMVC)*, 2008.

[30] J. Shi and J. Malik. Normalized cuts and image segmentation. *(PAMI)*, 2000.

[31] J. Shotton, J. Winn, C. Rother, and A. Criminisi. Textonboost: Joint appearance, shape and context modeling for multi-class object recognition and segmentation. In *(ECCV)*, 2006.

[32] J. Shotton, M. Johnson, and R. Cipolla. Semantic texton forests for image categorization and segmentation. In *(CVPR)*, 2008.

[33] P. Sturgess, K. Alahari, L. Ladicky, and P.H.S. Torr. Combining appearance and structure from motion features for road scene understanding. In *(BMVC)*, 2009.

[34] A. Torralba, K. P. Murphy, W. T. Freeman, and M. A. Rubin. Context-based vision system for place and object recognition. In *(ICCV)*, 2003.

[35] A. Torralba, K. P. Murphy, W. T. Freeman, and M. A. Rubin. Sharing features: Efficient boosting procedures for multiclass object detection. In *(CVPR)*, 2004.

[36] J. M. Winn and J. Shotton. The layout consistent random field for recognizing and segmenting partially occluded objects. In *(CVPR)*, 2006.

[37] X. Yang, N. Adluru, and L. J. Latecki. Particle filter with state permutations for solving image jigsaw puzzles. In *(CVPR)*, 2011.