Efficient Spatial Classification using Decoupled Conditional Random Fields

Chi-Hoon Lee, Russell Greiner, and Osmar Zaïane

Department of Computing Science, University of Alberta, Edmonton, AB, Canada

Abstract. We present a discriminative method to classify data that have interdependencies in 2-D lattice. Although both Markov Random Fields (MRFs) and Conditional Random Fields (CRFs) are well-known methods for modeling such dependencies, they are often ineffective and inefficient, respectively. This is because many of the simplifying assumptions that underlie the MRF's efficiency compromise its accuracy. As CRFs are discriminative, they are typically more accurate than the generative MRFs. This also means their learning process is more expensive. This paper addresses this situation by defining and using "Decoupled Conditional Random Fields (DCRFs)", a variant of CRFs whose learning process is more efficient as it decouples the tasks of learning potentials. Although our model is only guaranteed to approximate a CRF, our empirical results on synthetic/real datasets show that DCRF is essentially as accurate as other CRF variants, but is many times faster to train.

1 Introduction

Much of data mining deals with ways to learn classifiers from data samples. While many standard learning systems (*e.g.*, SVM, Logistic Regression, Naïve Bayes, Decision Trees, etc.) are designed to deal with independent and identically distributed data, this paper deals with interdependent data — viz., classifying regions in a 2-D lattice. In particular, we consider the task of detecting and delimiting tumors in Magnetic Resonance (MR) images of a patient's brain, which involves labeling each pixel as either tumor or non-tumor. Since most tumors are contiguous regions, we expect the labels of spatially adjacent pixels to belong to the same class, assuming they have sufficiently similar features.

Many effective region classifiers incorporate spatial constraints to encode the fact that the labels of neighboring pixels are typically correlated. In particular, there are a number of "random field" approaches for such tasks, including generative models like Markov Random Field (MRF) [13, 9], as well as discriminative models, including Conditional Random Field (CRF) [11] and its variants — Discriminative Random Field (DRF) [10], Associative Markov Nets (AMN) [19] , and our recent Support Vector Random Field (SVRF) [12]. As an MRF assumes conditional independence among observations given class labels, their learning procedures tend to be faster than the discriminative models (variants of CRFs); however, this assumption means they are typically not as accurate. The more accurate models, unfortunately, can be prohibitively slow to train, which may not be tolerable to a data mining task. We therefore propose a novel variant to our discriminative random fields model to make them more efficient

to train: we develop a "decoupled" learner, DCRF that reduces the expense of learning the random fields. We found that, as expected, the resulting DCRF is much faster to train than other CRF variants. Moreover, we were pleasantly surprised to find that this improvement in speed did not cost a degradation in accuracy!

Section 2 presents a quick overview of related systems. It motivates our approach by noting that these related systems – especially the ones that produce accurate labelings – can be very slow to train. Section 3 introduces our novel "Decoupled Conditional Random Field" (DCRF) approach, and provides algorithms for both learning the parameters and for inference (*i.e.*, classification — here segmentation). Section 4 demonstrates the accuracy and efficiency of our model by presenting experimental results over various domains, including the challenging real-world problem of segmenting brain tumor from MRI scans.

2 Related Work

There are now many systems for learning the spatial correlations; this paper focuses on ones based on random fields.

An Markov Random Field (MRF) is a *generative* approach that models the joint probability distributions over a set of instances $\mathbf{x} = \langle x_i \rangle$ (where each x_i corresponds to a vector of values describing the i^{th} pixel) and their associated class labels $\mathbf{y} = \langle y_i \rangle$. As with other random fields, a MRF provides a form for computing $P(\mathbf{y} \mid \mathbf{x})$, based on both properties of each instance (*i.e.*, "pixel") as well as features of their "neighbors" (*i.e.*, "properties and perhaps labels of adjacent pixels"), towards returning the most likely $\mathbf{y}^*(\mathbf{x}) = \operatorname{argmax}_{\mathbf{y}} P(\mathbf{y} \mid \mathbf{x})$.

In the MRF framework, the posterior over the n joint labels $\mathbf{y}$ given the observations $\mathbf{x}$ is $P(\mathbf{y} \mid \mathbf{x}) \propto P(\mathbf{y}) P(\mathbf{x} \mid \mathbf{y}) = P(\mathbf{y}) \prod_i^n P(x_i \mid y_i)$. Estimating the likelihood is computationally tractable as it is factored as $P(\mathbf{x} \mid \mathbf{y}) = \prod_i P(x_i \mid y_i)$. As this factorization is only a crude approximation to reality, this approach will typically produce inferior labels. The prior $P(\mathbf{y})$ can explicitly incorporate dependencies among the labels. Considering the equivalence between MRF and Gibbs Distributions [1], the posterior is formulated as

$$P(\mathbf{y} \mid \mathbf{x}) = \frac{1}{Z(\mathbf{x})} \exp\left[\sum_{c \in C} V_c(\mathbf{y}_c) + \sum_{i \in S} \log(P(x_i \mid y_i))\right] , \tag{1}$$

where C is a set of cliques defined in 2-D lattice.$V_c(\mathbf{y})$ is a clique potential function of labels for clique $c \in C$, S is the set of nodes (*i.e.*, pixels), and the "partition function" $Z(\mathbf{x}) = \sum_{\mathbf{y}'} \exp\left[\sum_{c \in C} V_c(\mathbf{y}'_c) + \sum_{i \in S} \log(P(x_i \mid y'_i))\right]$ is used to normalize the resulting values. Notice $V_c(\mathbf{y}_c)$ depends only on the labels $\{y_i\}$, but not on the information about the pixels $\{x_i\}$. Therefore, a MRF prefers a set of labels $\mathbf{y}^*$ where neighbors have the same value [1, 13], independent of properties of these pixels. Also, as the partition function $Z(\mathbf{x})$ involves summing over all $|L|^n$ possible labelings (assuming there are $|L|$ labels for each pixel), it is very expensive to compute the exact value of the partition function.

A discriminative model, Conditional Random Field (CRF) [11], attempts to overcome the disadvantages of a MRF — notably its conditional independence assumption

and the absence of observation information in the second potential — by directly modeling the posterior distribution $P(\,\mathbf{y}\,|\,\mathbf{x}\,)$ as

$$P(\,\mathbf{y}\,|\,\mathbf{x}\,) = \frac{1}{Z(\mathbf{x})} \exp\left(\sum_{i\in S}\left[\Phi_{\mathbf{w}}(y_i,\mathbf{x}) \;+\; \sum_{j\in N_i}\Psi_{\nu}(y_i,y_j,\mathbf{x})\right]\right) \tag{2}$$

which directly computes a posterior distribution without modeling the prior $P(\,\mathbf{y}\,)$. The notation is essentially the same as in Equation 1: $Z(\mathbf{x})$ is the partition function, S is the set of pixels in an image, $\mathbf{x} = \langle\, x_i \,\rangle$ is the set of descriptions of those pixels, and $\mathbf{y} = \langle\, y_i \,\rangle$ is the set of labels. Here N_i is the set of neighbors of node x_i — in 2D, the pixel at location (a, b) has 4 neighbors, at $(a-1, b)$, $(a+1, b)$, $(a, b-1)$ and $(a, b+1)$ [1, 9]. For notation, "$\Phi_{\mathbf{w}}(y_i, \mathbf{x})$" is called the "Association" potential, which deals with a single instance. While its value can depend on all of $\mathbf{x}$, it typically relies only on x_i. The "$\Psi_{\nu}(y_i, y_j, \mathbf{x})$" term is called the "Local-Consistency" (or "Interaction") potential in variants of CRF such as SVRFs and DRFs; it is typically used to prefer labeling that assign the same class labels to neighboring pixels. (We can view $\Psi_{\nu}(\cdot)$ as a data dependent smoothing function; this differs from a MRF, which instead use only a "data independent" term.) Here, $\mathbf{w}$ and ν refer to the parameters associated with these potential functions.

Note that a CRF and its variants — DRFs and SVRFs— typically produce better accuracy than their generative alternative MRF. However, their good performance comes at a cost: the learning process is significantly more expensive. For example, the learning task in DRF and SVRF involves estimating the parameters $\mathbf{w}$ and ν that maximize the (log)likelihood of the given data sample, and both systems use a regularization term to avoid overfitting. The log-likelihood is formulated as

$$\langle\, \hat{\mathbf{w}}, \hat{\nu}\,\rangle = \underset{\mathbf{w},\nu}{\operatorname{argmax}} \left\{ \sum_{k=1}^{M}\sum_{i=1}^{S} \Phi_w(y_i,\mathbf{x}) + \sum_{j\in N_i} \Psi_{\nu}(y_i^{(k)}, y_j^{(k)}, \mathbf{x}^{(k)}) - \log(Z^{(k)}(\mathbf{x})) \right\} - \frac{\nu^T\nu}{2\tau} \tag{3}$$

Although a SVRF can significantly improve the accuracy of a DRF, especially when features may be correlated, the study [12] has shown that selecting the appropriate τ in a SVRF and a DRF is a non-trivial task, which makes the learning procedures more challenging and costly. Associative Markov Nets (AMN) [19], which discriminatively train Markov nets, exploit the spatial correlations by adopting the maximum-margin principle of maximizing the margin between target labels and the best runner-up label assignments. Hence, this process employs the same ideas underlying SVMs. (Note that our SVRF differs by actually performing the same basic computations that an SVM performs.) A Boosted Random Field (BRF) [21] combines the set of iid classifiers that correspond to Association potentials. Each potential in a BRF is trained on a specific class to quantify the likelihood of a class on a pixel. Hence, BRF does not explicitly consider the spatial correlation.

We see there are problems in training each of the systems mentioned in this section: some are inaccurate (as they use inappropriate models), while others require too much computation time.

3 The DCRF System

This section presents the foundations to formalize our Decoupled Conditional Random Field, DCRFs of random fields. We first motivate our approach of decoupling the training of the two potentials, then discuss inference — *i.e.*, how to use the resulting system to classify pixels in an image.

First, if we ignore the dependencies among the labels of the pixels (*i.e.*, assume that they are independent and identically distributed), we would use only the "Association" potential, which attempts to maximize

$$P_A(\,\mathbf{y}\,|\,\mathbf{x}\,) \quad \propto \quad \exp\left(\sum_{i\in S} \Phi(y_i, \mathbf{x})\right) \tag{4}$$

Many existing classifiers (*e.g.*, Naïve Bayes, Logistic Regressions, SVM, etc.) are (perhaps implicitly) attempting to optimize Equation 4.

Alternatively, a discriminative model that only considers spatial coherence would attempt to optimize

$$P_{LC}(\,\mathbf{y}\,|\,\mathbf{x}\,) \quad \propto \quad \exp\left(\sum_{i\in S} \Psi(y_i, y_{N_i}, \mathbf{x})\right) \tag{5}$$

where y_{N_i} are the labels of i's neighbors.

Equation 4 and 5 provide different frameworks for approximating the posterior probability distributions $P(\,\mathbf{y}\,|\,\mathbf{x}\,)$. Each is only partial, in that the first (second) does not properly incorporate spatial coherence (resp., the local observations).

Notice typical CRF models involve the sum of these equations — written in log space as

$$\sum_{i\in S} \Phi(y_i, \mathbf{x}) \quad + \quad \sum_{i\in S} \Psi(y_i, y_{N_i}, \mathbf{x}) \tag{6}$$

(Compare to Equation 2. Note that the neighborhood is considered in $\Psi(\cdot)$ explicitly.)

We now observe that each classifier form in Equation 6 follows MAP formulations for the joint probability over labels: that is, we can approximate the global optimal joint class labels by maximizing the local posterior probability distribution using the principles of pseudo-likelihood and Iterative Conditional Modes (ICM)[1] [3] — *i.e.*, $P(\,\mathbf{y}\,|\,\mathbf{x}\,) = \prod_{i\in S} P(\,y_i\,|\,y_{N_i}, \mathbf{x}\,)$. Thus, for each pixel i, the log of ensemblized posterior distribution $P(\,y_i\,|\,y_{N_i}, \mathbf{x}\,)$ given its neighbors y_{N_i} is:

$$\Phi_{\mathbf{w}}(y_i, \mathbf{x}) \; + \; \sum_{j\in N_i} \Psi_\nu(y_i, y_j, \mathbf{x}) \tag{7}$$

[1] Pseudo-likelihood and ICM are only guaranteed to achieve local maxima, the discussion of the global optimality issues is beyond the scope of this paper.

N.b., as we will only be seeking the argmax, we do not need to consider the normalizing "$-\log(z_i)$" term that shown in Equation 3, as it will be constant here.

Equation 7 shows that we can approximate a CRF model using a decoupled system, corresponding to the simple sum of two different potentials, *which are learned separately*. (This differs from standard ensemble methods [5], as we are directly combining *potentials* rather than classifiers.) However, there is one remaining question: how to deal with the relative scaling issues when combining of the two potentials. This will be discussed in the following sections. We will also see that, as expected, it is much faster to learn these individual summands *individually*, before combining them. Our empirical evidence shows that, surprisingly, the resulting DCRF system can be as accurate!

Association-only Potential The association potential provides a local posterior for each pixel: $P_A(\, y_i \,|\, x_i \,)$. Our "decoupling" principle allows us to select a function that quantifies the conditional probability for a given observed instance. We incorporate a maximal margin approach where the two classes of pixels are classified based on a hyperplane that maximizing the distances between the two classes.

As suggested above, we will consider a potential based on SVMs; note this method inherits the SVM's relative insensitivity to class imbalance, and its ability to typically outperform other discriminative classifiers such as GLMs, especially in cases where the classes overlap [18], which is common case in imaging applications.

We select the hyperplane by solving the following optimization problem (over the t instances):

$$\begin{aligned} &\max_{w,b,\gamma} \quad \gamma \\ &\text{subject to} \quad y_i(w^T x_i + b) \geq \gamma, i = \ 1, \cdots, t \ ; \ \ \|w\|^2 = 1 \end{aligned} \tag{8}$$

where γ is a margin, b is a bias term, and the vector w is normal to a hyperplane that we are seeking, which separates the positive from the negative examples. Using its dual formulation with dual variables α_i, we solve this optimization problem using Quadratic Programming (QP) over the α_is, to produce $f(x) \quad = \quad \sum_{i=1}^{t} \alpha_i \, y_i \, x_i^T \, x \ + \ b$ then use the decision function *sign* $(\, f(x) \,)$ to classify the test instance x. Note this learning process requires only polynomial time. Our implementation actually uses Sequential Minimal Optimization (SMO) [15], which is an efficient implementation.

Notice that $f(x)$ computes the distance to the hyperplane from the instance x. We can use this to compute (something like) a posterior probability function [16, 14]:[2]

$$\Phi_{\mathbf{w}}(y_i, \mathbf{x}) \ = \ \frac{1}{1 + \exp(A_A \times y_i(\mathbf{w}^T x_i) + B_A)} \tag{9}$$

using the parameters A_A and B_A that optimize the fit of the training data to a sigmoid function [16, 12].

[2] Of course, we augment the instance x_i by including a constant 1, and hence the $\mathbf{w}$ include a "constant" term as well.

Local-Consistency-only Potential We use our "local-consistency-only" potential to model the "neighborhood coherence" between pixels. Its goal is to encourage "similar" instances within the specified each neighborhood to have the same labels. Although we can use the associated potential as a stand-alone decision function, its function here is mainly to smooth regions (and hence remove errors) produced by the Association-only potential.

For similar instances in a neighborhood to have similar (in our discrete case, "identical") class labels, we introduce a max-margin based potential, which tries to make the labels of a testing instance the same as the labels of its neighbors. This potential learns a pairwise max-margin model that quantifies the likelihood that two pixels will have the same class labels, given their descriptions:

$$\Psi_\nu(y_i, y_j, \mathbf{x}) \;=\; I(y_i, y_j) \times [\nu^T \langle\, \psi(x_i, x_j)\,\rangle] \tag{10}$$

where $I(y_i, y_j)$ returns $+1$ if $y_i = y_j$, and -1 otherwise. (We define $\psi(x_i, x_j)$ below.) Equation 10 reduces the pairwise discriminative learning problem to the binary class problem, over similar versus dissimilar classes. That is, we apply QP to the training set

$$S_{new} = \quad \{\, (\psi(x_r, x_j),\; I(y_r, y_j)) \mid j \in N_r \,\}$$

over all instances r with neighbors $j \in N_r$, to find the optimal parameter ν.

Note that each pair of pixels is projected by $\psi(\cdot)$ onto a similarity feature space. For instance, we could use $\psi(x_i, x_j) = x_i^T x_j$ that produces a scalar: the cosine measure of the similarity. Note this attains its largest value as the two vectors match one another. Due to "localized" neighborhood system we consider for Local-consistency potential, the increment only grows linearly with the number of pixels. Notice that feature-wise space depends on $\psi(\cdot)$.

As we will need to combine this potential with the Association-only one, we need to produce values within a "comparable" range. We therefore convert Equation 10 to the posterior probability scale, using the same transformation used to produce Equation 9.

$$\Psi(y_i, y_j, \mathbf{x}) = \frac{1}{1 + \exp\left(A_{LC} \times I(y_i, y_j)(\nu^T \langle\, \psi(x_i, x_j)\,\rangle) + B_{LC}\right)} \tag{11}$$

where again A_{LC} and B_{LC} are set to optimize the fit to a sigmoid, which produces a probability distribution as in Association-only potential

3.1 Inference

Our goal in producing this DCRF system is then to find relevant regions within images — *e.g.*, tumor regions within MR images of a brain. This involves inferring a binary label (tumor versus non-tumor) for each individual pixel. As noted above, this corresponds to computing the most likely vector $\mathbf{y}^* = \mathrm{argmax}_{\mathbf{y}}\, P(\,\mathbf{y} \mid \mathbf{x}\,)$ given the evidence vector $\mathbf{x}$, based on the (possibly unnormalized) potential functions. In our case, we will use the potential functions in Equation 7, which is the sum of the Association-only $P_A(\,\cdot\,)$ (Equation 4) and Local-Consistency-Only $P_{LC}(\,\cdot\,)$ (Equation 5) potentials. While this exact computation can be expensive, there are several existing approximation algorithms for CRF, including Iterative Conditional Modes (ICM) [3], Graph-Cuts

(GC) [2], and Loopy Belief Propagation (LBP) [7]. DCRF uses ICM since it converges quickly and has been shown empirically to produce accurate results [12, 1].[3] Also, while ICM may converge to local optima for the joint distribution problem, it works sufficiently well by iteratively propagating the belief for each pixel to its neighboring pixels:

$$y_i^* = \underset{y_i \in \{+1,\,-1\}}{\operatorname{argmax}} P(\, y_i \,|\, y_{N_i}, \mathbf{x}\,) = \underset{y_i \in \{+1,\,-1\}}{\operatorname{argmax}} \Phi(y_i, \mathbf{x}) \;+\; \sum_{j \in N_i} \Psi(y_i, y_j, \mathbf{x}) \quad (12)$$

Of course, we could add the normalization factor z_i in Equation 12, which constrains outputs to follow probability axioms. However, the constant factor is irrelevant, since our inference approach seeks only the most likely value.

3.2 Complexity

Our DCRF model uses Quadratic Programming to learn the parameters, within SMO. Assuming each image has S pixels, and each pixel has E neighbors then learning the Association-only potential requires $O(S^2)$ steps per image, and Local-Consistency-only potential requires $O(\,(S \times E)^2\,)$ per image. Note that in our paper, we used E is 4.

Inference (here, classifying the regions in a test image) requires $O(\,S + \,(S \times E)\,)$ per iteration. Empirically, we found that ICM converged after 5 iterations, on average.

4 Experiments

We implemented the Decoupled CRF described above, DCRF, and compared it with other random field techniques on both synthetic and real-world tasks. As many imaging tasks are very imbalanced (in that the "positive" class includes only a small percentage of the pixels), the standard evaluation criteria of "accuracy" is problematic. We therefore use the Jaccard score — $J = \frac{TP}{TP+FP+FN}$ — to measure its performance, using true positives (TP), false positives (FP), and false negatives (FN).

Synthetic image sets The primary goal in using the synthetic data sets is to see how the various algorithms segment objects in the presence of noise. We therefore evaluated these techniques over 15 synthetic image sets, each with its own shape, whose intensities were each independently corrupted by noise generated from $\mathcal{N}(0, 1)$. Here each image is of size 64-by-64 (4096 pixels). Note that some of image sets are significantly imbalanced, while others are balanced.

Figure 1 shows some of the experiment results. All Jaccard scores and elapsed learning times that appear are averaged over 3-fold cross-validation. Each row in Figure 1

[3] While GC and LBP are often considered be the best inference methods, even if the graph structure has loops, we used ICM for the reasons shown above. Note this issue is orthogonal to the goal of this paper, which is to compare the training time and accuracy of our DCRF to other CRF-related models.

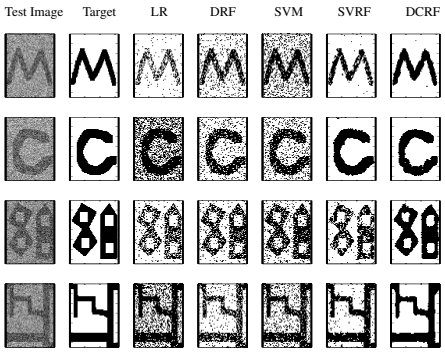

Fig. 1. Results from synthetic image sets. Rows 1 to 5 from the top down correspond respectively to datasets 7, 3, 10, and 11 in Fig. 2

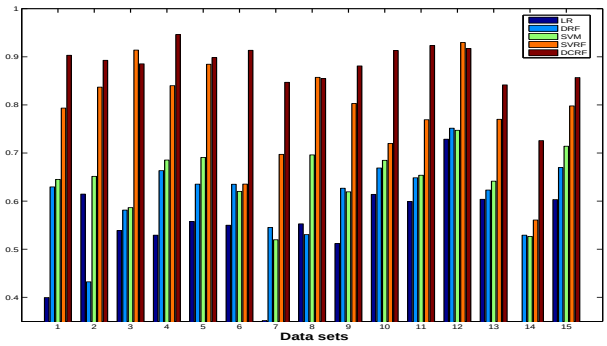

Fig. 2. Average Jaccard scores on 15 synthetic data sets

presents one example, showing (from left to right), the test images, the true labels, and outputs from Logistic Regression (LR), DRF, SVM, SVRF, and DCRF. We see that, overall, SVRF and DCRF are most accurate. Especially when the test images are imbalanced (the first row in Fig. 1), LR (third column) and DRF (fourth column) produce degraded outputs caused by the poor parameter estimations from the imbalanced data.

The second row illustrates the sensitivity of the regularization term τ in the SVRF frameworks. Although the correct value for this parameter can produce good segmentation results, in general it is not trivial to find such "good" values. While we can use cross-validation method to estimate this parameter, others [12, 10] have shown that this does not guarantee acceptable performance. Also, note that SVM-based approaches appear robust to the class imbalance, as empirically shown in [12].

Figure 2 shows that DCRF and SVRF are the two best performers overall, at this segmentation task, dealing with both the balanced and imbalanced data: Each was significantly better than the others at the $p < 1.14\text{E-}12$ level based on a paired example

Table 1. Average elapsed learning time (seconds)

	DRF	SVRF	DCRF
Synthetic	1581.3	714.5	21.2
Brain Tumor	1392.4	1209.4	82.3

t-test; moreover, DCRF performs better than SVRF at the $p < 0.0037$ level. Note that SVRF can sometimes produce better results than DCRF — see data sets 3, 9, and 12 in Figure 2. Here, we assume that SVRF found good estimates for τ. Data sets 6, 7, 9, 12, and 14 show that good estimation of the regularization in DRF performs better than SVM.

The first row of Table 1 reports the average learning time for DRF, SVRF and DCRF over these 15 cases. Notice first that our DCRF requires *significantly* less time than the other two approaches — 30 times faster than SVRF (and so $p <$ 1.165E-17) and over 70 times faster than DRF. This is because there are fast ways to solve DCRF's underlying QPs which we attribute to the observation that the SVRF learner regards the Association potential as a constant while learning the Local consistency potential, but a DRF attempts to optimize both potentials simultaneously. (Note this is more than compensates for the fact that DRF's Logistic Regression learning, by itself, would be faster than SVRF's QP.) Finally, recall that DCRF does not compute the partition function during the training.

Real-world problems We next applied these various learners to the task of segmenting brain tumors from MR images. Tumor segmentation is challenging for many reasons, including the differences between the brains of different individuals, and the fact that the same intensity values can be a tumor in one part of the image, but normal tissue in another [6, 8]. Automatic tumor segmentation would be very useful, as it would enable radiation oncologists to effectively locate the tumor, with sufficient precision that they can use this to perform diagnosis and to plan treatments.

Our experimental data sets consists of 13 volumes taken from 7 patients, each having either a grade 2 astrocytoma, an anaplastic astrocytoma, or a glioblastoma multiform. We focused only on the axial MRI slices — there were around 21 such slices per patient-visit. For each slice, there are three complete images, corresponding to three standard modalities, called "t1", "t2" and "t1c" [8]. These represent challenging cases since the tumor area is typically heterogeneous.

We used the multi-scale feature set based on [17], which contains traditional image-based features in addition to three types of 'alignment-based' features: spatial probabilities for each of the 3 normal tissue types (white matter, gray matter, CSF), spatial expected intensity maps, and a characterization of left-to-right symmetry; each measured at multiple scales. As with many of the related works on brain tumor segmentation (such as [6, 22]), our training is a patient-specific scenario, where training data for the classifier is obtained from the patient to be segmented. Note that pixels to be tested are from a brain slice that is different from the slice containing the training pixels.

In our experiment, we evaluated the following 7 classifiers on the 13 different time points from the 7 patients brain volumes. Maximum Likelihood (ML $\equiv$ degenerate MRF), Logistic Regression (LR $\equiv$ degenerate DRF), SVM (degenerate SVRF), MRF, DRF, SVRF and DCRF. For each of the Random Field methods, we initialized inference

Table 2. Jaccard Percentage Scores for Enhancing Tumor and Edema Tumor Areas.

	Enhancing Tumor Area							Edema Area						
Studies	ML	MRF	LR	DRF	SVM	SVRF	DCRF	ML	MRF	LR	DRF	SVM	SVRF	DCRF
1-1	23.1	24.6	44.4	46.1	50.7	52.8	**53.2**	21.9	21.6	35.7	36.7	58.0	**58.2**	58.0
2-1	0.0	0.0	61.3	61.5	87.4	**87.7**	87.1	33.3	34.2	59.2	61.4	**89.4**	89.2	89.3
3-1	69.2	69.7	61.8	61.8	83.0	84.8	**86.8**	34.4	34.4	75.5	77.2	81.7	**82.2**	81.9
3-2	40.1	40.3	84.8	84.6	85.7	**85.8**	**85.8**	47.6	48.1	73.6	74.1	80.3	**81.1**	80.5
4-1	26.9	27.3	49.1	50.4	78.8	81.7	**82.6**	28.3	29.1	38.6	41.2	54.0	**55.4**	54.6
4-2	58.9	59.7	68.3	70.2	76.7	77.9	**79.2**	43.2	46.8	45.3	46.7	54.7	**57.7**	54.9
4-3	49.2	50.2	71.3	71.6	88.2	88.1	**88.8**	35.4	35.4	69.9	**70.6**	69.2	69.1	69.1
4-4	65.6	68.2	**87.5**	87.1	87.0	87.1	86.9	44.1	43.7	78.6	79.0	77.7	77.3	**79.5**
5-1	67.0	67.5	52.2	51.4	82.8	**84.3**	84.1	47.8	48.6	63.6	65.7	74.8	**76.9**	74.6
6-1	37.4	37.6	76.4	76.2	79.2	**80.4**	80.0	40.3	40.1	79.3	79.7	82.2	**83.7**	82.9
7-1	63.2	63.0	75.5	76.7	81.0	**81.4**	81.1	74.9	77.7	91.2	92.4	94.8	**94.9**	**94.9**
7-2	37.7	39.3	75.9	75.8	86.5	**87.3**	86.8	39.2	40.4	80.9	82.7	**83.1**	82.8	**83.1**
7-3	45.3	45.6	81.8	81.5	87.7	87.6	**87.8**	54.1	53.9	79.3	80.7	84.6	84.5	**85.6**
Average	44.9	45.6	63.6	68.8	81.1	82.1	**82.3**	41.9	42.6	62.2	68.3	75.7	**76.4**	76.1

with the corresponding degenerate classifier (*i.e.*, Maximum Likelihood, Logistic Regression, or SVM). To provide a fair comparison between SVM-based models (SVRF and DCRF) and the other models, we only used the linear kernel.

The first task was the relatively easy one of segmenting the "enhancing" tumor areas — the region that appears hyper-intense after injecting a contrast agent. The second task was segmenting the entire edema area associated with the tumor; this is significantly more challenging due to the high degree of similarity between the intensities of edema areas and normal cerebrospinal fluid in the various modalities. The final task was segmenting the gross tumor area as defined by the radiologist. This can be a subset of the edema but a superset of the enhancing area, and is inherently a very challenging task even for human experts, given the modalities examined.

Tables 2 and 3 present the classification results for the three tasks. Over all three tasks, we see that the best results were typically obtained by either DCRF and SVRF, which were comparable to each other, and statistically better than the rest: The differences between SVRF and the next best, SVM, across the three tasks was significant at the $p < 0.000002$ level based on a paired example t-test, but the same t-test between SVRF and DCRF across the tasks indicates no difference — *i.e.*, here $p = 0.37$. However, Table 1 (second row) shows that our method requires significantly less training time — by a factor of 14! ($p <$ 2.285E-34) Although SVM performed very well visually on the three tasks, just as we saw on the synthetic data results, this performance can not always be guaranteed. In Table 2, the results from the second patient "2-1" produced an interesting observation; significant overlap between Gaussians in the high dimensional feature space leads ML and subsequently MRF to misclassify the entire area as non-tumor. This example shows that inappropriate modeling of $P(\mathbf{x} \mid \mathbf{y})$ can generate extremely poor performance. Although the segmentation tasks for edema and gross tumor areas are very hard, the best discriminative approaches (*i.e.*, SVRF and DCRF) still produce segmentations that are typically very similar to the manual segmentations, for all 3 tasks.

Table 3. Jaccard scores for Gross Tumor Areas.

	Gross Tumor Area						
Studies	ML	MRF	LR	DRF	SVM	SVRF	DCRF
1-1	19.3	19.5	39.4	**40.9**	40.7	40.5	41.1
2-1	35.4	35.7	65.1	66.1	**78.2**	76.9	78.0
3-1	44.4	46.1	72.9	73.4	77.9	78.7	**78.2**
3-2	51.2	51.3	76.3	76.2	78.1	78.8	**80.2**
4-1	37.4	38.7	39.4	40.1	41.4	41.2	**42.1**
4-2	38.0	40.2	39.7	39.4	62.1	**64.9**	62.1
4-3	66.0	68.5	73.3	**73.5**	64.4	64.5	64.1
4-4	46.7	45.8	83.8	83.5	86.0	**87.0**	86.2
5-1	50.1	50.9	65.3	68.3	82.8	**84.8**	83.4
6-1	46.6	47.6	79.6	79.4	87.6	**88.2**	87.8
7-1	66.4	66.3	71.9	73.2	74.6	74.1	**74.7**
7-2	49.6	52.4	68.3	67.9	72.7	**72.9**	72.5
7-3	43.4	43.7	73.5	72.7	81.6	81.2	**82.0**
Average	45.7	46.7	60.6	65.7	71.4	**71.8**	71.7

5 Conclusions

Learning to classify regions in an image is a challenging task, partly because labeling each pixel in an image can require modeling spatial correlations among neighboring pixels, which can be difficult to learn. As standard independent and identically distributed classification algorithms do not model these correlations, they typically fail to correctly classify data instances. Such spatial correlations can, however, be effectively modeled by various Random Field frameworks. However, these systems (especially the ones that work effectively.) can require a significant amount of time to learn. This time constraint makes such models inappropriate for large scale real-world problems, such as segmenting brain tumors.

In this paper, we have proposed a Decoupled CRF (DCRF) to improve the efficiency of a discriminative Random Field method for finding regions in an image. Our proposed model first learns the two potentials (Association and Local-consistency) *independently*, each based on a variant of Support Vector Machines. Afterwards, to segment regions in a novel image, it uses a new potential that is the simple sum of these potentials, using ICM (with respect to this combined potential) to produce a labeling. Our empirical results — on both synthetic and real-world data — show that this DCRF approach is virtually as accurate as the most accurate random field for this task (SVRF), but the learning time is many times faster (here, by a factor over 14 in one case, and over 30 in another). In addition, our model produces effective classification results, even when data sets are heavily imbalanced.

Acknowledgments R. Greiner is supported by the National Science and Engineering Research Council of Canada (NSERC) and the Alberta Ingenuity Centre for Machine Learning (AICML). C.H. Lee is supported by AICML. O. Zaïane is supported by NSERC. Our thanks to Dale Schuurmans for helpful discussions on optimization and

parameter estimation, BTGP members for help in data processing, and Albert Murtha (M.D.) for domain knowledge on the tumor data set.

References

1. J. Besag. On the statistical analysis of dirty pictures. *Journal of Royal Statistical Society. Series B*, 48:3:259–302, 1986.
2. Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *ICCV*, pages 377–384, 1999.
3. C. J. C. Burges. A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery*, 2(2):121–167, 1998.
4. T. Chen and D. N. Metaxas. Gibbs prior models, marching cubes, and deformable models: A hybrid framework for 3d medical image segmentation. In *MICCAI*, pages 703–710, 2003.
5. T. G. Dietterich. Machine-learning research: Four current directions. *The AI Magazine*, 18(4):97–136, 1998.
6. C. Garcia and J. Moreno. Kernel based method for segmentation and modeling of magnetic resonance images. *LNCS*, 3315:636–645, Oct 2004.
7. M. I. Jordan. *editor. Learning in Graphical Models*. MIT Press, 1999.
8. M. Kaus, S. Warfield, A. Nabavi, P. Black, F. Jolesz, and R. Kikinis. Automated segmentation of MR images of brain tumors. *Radiology*, 218:586–591, 2001.
9. R. Kindermann and J. Snell. Makrov random fields and their applications. *American Mathematical Society*, 1980.
10. S. Kumar and M. Hebert. Discriminative fields for modeling spatial dependencies in natural images. *NIPS*, 2003.
11. J. Lafferty, F. Pereira, and A. McCallum. Conditional random fields: Probabilistic models for segmenting and labeling sequence data. *ICML*, 2001.
12. C. Lee, M. Schmidt, and R. Greiner. Support vector random fields for spatial classification. *PKDD*, 2005.
13. S. Z. Li. *Markov Random Field Modeling in Image Analysis*. Springer-Verlag, Tokyo, 2001.
14. H.-T. Lin, C.-J. Lin, and R. C. Weng. A note on platt's probabilistic outputs for support vector machine. Technical report, 2003.
15. J. Platt. Fast training of support vector machines using sequential minimal optimization. In *Advances in Kernel Methods - Support Vector Learning*, pages 185–208. MIT Press, 1999.
16. J. Platt. *Probabilistic outputs for support vector machines and comparison to regularized likelihood methods*. MIT Press, Cambridge, MA, 2000.
17. M. Schmidt. Automatic brain tumor segmentation. Master's thesis, University of Alberta, 2005.
18. J. Shawe-Taylor and N. Cristianini. *Kernel Methods for Pattern Analysis*. Cambridge University Press, Cambridge, UK, 2004.
19. B. Taskar, V. Chatalbashev, and D. Koller. Learning associative markov networks. In *ICML '04*, page 102, New York, NY, USA, 2004. ACM Press.
20. B. Taskar, C. Guestrin, and D. Koller. Max margin markov networks. In *NIPS*, 2003.
21. A. Torralba, K. P. Murphy, and W. T. Freeman. Contextual models for object detection using boosted random fields. In *NIPS 17*. MIT Press, Cambridge, MA, 2005.
22. J. Zhang, K. Ma, M. Er, and V. Chong. Tumor segmentation from magnetic resonance imaging by learning via one-class support vector machine. *International Workshop on Advanced Image Technology*, pages 207–211, 2004.